

# **BAB I**

---

## **E-COMMERCE DALAM PENJUALAN PRODUK BERBASIS WEBSITE**

---

### ***1.1 E-Commerce***

### 1.1.1 Definisi E-Commerce



Secara umum, istilah E-commerce (perdagangan elektronik) adalah aktivitas pembelian dan penjualan barang / jasa atau pengiriman dana / data melalui jaringan elektronik, khususnya internet. Dengan perkembangan teknologi informasi dan perangkat lunak, ini telah memungkinkan transaksi elektronik konvensional. Situs web ini digunakan sebagai aplikasi toko offline. Situs web e-commerce mengelola berbagai fungsi seperti tampilan produk, pemesanan online, dan inventaris stok, untuk menjalankan fungsi utama sebagai e-commerce. Perangkat lunak yang digunakan diinstal pada server e-commerce dan digunakan bersama dengan sistem pembayaran online untuk pemrosesan transaksi. Secara umum e-commerce berarti berbisnis melalui jaringan yang saling berhubungan (network / internet interconnection).

Aktivitas menggunakan informasi digital dan teknologi pemrosesan komunikasi dalam melakukan transaksi bisnis untuk menciptakan, mengubah, dan mendefinisikan kembali hubungan antara penjual dan pembeli adalah sebuah definisi lain dari E-commerce. Sederhananya, E-commerce dapat diartikan sebagai transaksi pembelian dan penjualan barang, layanan atau transfer dana atau data menggunakan elektronik yang terhubung ke internet. Transaksi e-commerce tidak lagi baru di negara ini, bahkan perkembangannya cukup cepat.

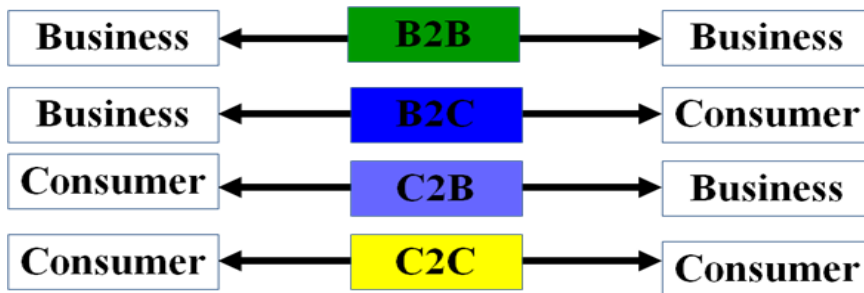
### 1.1.1 Sejarah E-Commerce



E-Commerce pertama kali diperkenalkan pada tahun 1994 ketika spanduk elektronik pertama kali dibeli untuk tujuan promosi dan iklan di halaman web (situs web). Menurut Forrester Research, perdagangan elektronik menghasilkan penjualan US \$ 12,2 juta pada tahun 2003. Istilah "perdagangan elektronik" telah mengubah persaingan dengan waktu. Awalnya, perdagangan elektronik, sarana komersial, seperti penggunaan EDI untuk pembelian dokumen komersial, seperti pesanan pembelian atau faktur elektronik. Kemudian berkembang menjadi sebuah kegiatan yang memiliki istilah "web commerce" yang lebih tepat - pembelian barang dan jasa melalui World Wide Web melalui server yang aman (HTTPS), yaitu protokol khusus yang menggunakan enkripsi untuk data yang penting bagi pelanggan. Pada awalnya kompilasi web menjadi populer pada tahun 1994, banyak wartawan memperkirakan bahwa e-commerce akan menjadi sektor ekonomi baru. Namun, hanya sekitar empat tahun kemudian protokol aman seperti HTTPS dipasang dan digunakan secara luas. Kisaran antara 1998 dan 2000 banyak bisnis di AS dan Eropa mengembangkan situs web perdagangan ini.

### 1.1.2 Jenis-Jenis E-Commerce

Ada beberapa jenis transaksi dalam e-commerce, termasuk bisnis ke bisnis, bisnis ke konsumen, konsumen ke konsumen atau konsumen ke bisnis. Oleh karena itu, persyaratan e-commerce dan e-bisnis selalu disepakati.



#### 1. E-Commerce Bisnis ke Bisnis (B2B)

Transaksi e-commerce ini dilakukan oleh dua pihak yang memiliki kepentingan bisnis. Kedua belah pihak saling memahami dan mempelajari bisnis yang mereka jalankan.

Bisnis ini dijalankan secara berkelanjutan, atau hanya disetujui. Contoh sederhana B2B ini adalah produsen dan pemasok yang bertransaksi online baik untuk berkonsultasi dengan kebutuhan barang, hingga proses pembayaran.

#### 2. Bisnis E-commerce ke Konsumen (B2C)

Bisnis kepada konsumen dilakukan oleh bisnis dan konsumen. Transaksi e-commerce ini berlangsung seperti perdagangan normal. Pelanggan mendapatkan penawaran produk dan melakukan pembelian secara online.

#### 3. C2C e-commerce (Konsumen ke Konsumen)

Untuk C2C, transaksi dilakukan oleh konsumen ke konsumen. Jika Anda sering menggunakan Tokopedia, Bukalapak, OLX dan sejenisnya, maka inilah yang disebut B2C e-commerce.

Transaksi jual beli dilakukan secara online melalui pasar. Jadi C2C di sini adalah partisipasi antara penjual dan pembeli.

#### 4. Consumen to Busines (C2B)

C2B adalah kebalikan dari B2C di mana pelanggan terakhir bertindak sebagai penjual dan perusahaan bertindak sebagai pembeli.

#### 5. Media atau Aplikasi E-commerce

Sebagaimana dibahas dalam kasus e-commerce di atas, transaksi bisnis ini kompatibel dengan aplikasi media online lainnya (baca: media online), seperti katalog, email, keranjang belanja, layanan eb, EDI dan protokol transfer file. B2B (bisnis ke bisnis).

### ***1.2 Penjualan***



#### **1.1.2 Pengertian penjualan**

Penjualan adalah sebuah sumber kehidupan perusahaan, karena dari penjualan dapat diperoleh keuntungan dan bisnis menarik pelanggan yang berusaha mencari daya tarik, konsumen dapat mengetahui hasil dari produk yang dihasilkan. Promosi dilakukan untuk meningkatkan volume penjualan dan strategi untuk mendorong pembeli melakukan transisi. Sedangkan penjualan adalah suatu pemindahan barang dan jasa yang dilakukan atau dilaksanakan oleh penjual. Secara umum, perusahaan yang ingin menyelesaikan proses peningkatan volume penjualan akan melakukan kegiatan promosi melalui iklan, penjualan pribadi, dan publisitas. Istilah penjualan sering bingung dengan istilah promosi.

Sedangkan penjualan hanya mencakup penghapusan barang atau jasa, atau penggunaan penjual. Tidak Perlu Promosi. Jadi penjualan hanyalah bagian dari promosi. Sifat kegiatan dan struktur pelaporan dalam suatu perusahaan dapat berbeda. Namun, pada prinsipnya, semua itu dimaksudkan untuk memudahkan tugas penjualan. Untuk staf pemasaran, kegiatan pemasaran sangat penting. Mungkin salah satu yang paling penting adalah iklan, sementara efektivitasnya berhubungan langsung dengan kunjungan konsumen ke perusahaan. Dalam hal ini berkaitan langsung dengan jumlah penjualan yang dihasilkan. Dalam menjual masalah layanan yang sangat penting, kunjungan tenaga penjual ke pelanggan akan mempertimbangkan munculnya banyak pesanan. Fungsi tenaga penjualan adalah menghasilkan produk yang diproduksi oleh perusahaan. Selain itu bisa juga memberikan saran yang menarik. Ini sangat efektif untuk meningkatkan penjualan.

### 1.2.1 Sejarah Penjualan



Penjualan dalam praktiknya sudah ada sejak zaman peradaban kuno. Orang Yunani dan Romawi Kuno mempraktikkan ilmu perdagangan dan secara aktif berkomunikasi secara persuasif dengan konsumen mereka. Demikian juga di peradaban lain yang mempromosikan perdagangan. Namun, konsep pemasaran modern yang dikenal saat ini hanya muncul dan dikembangkan selama Revolusi Industri yang terjadi pada abad ke-18 dan ke-19. Periode ini ditandai oleh perubahan sosial yang didorong oleh perkembangan teknologi dan inovasi pengetahuan. Salah satu perubahan ini adalah industri yang memproduksi barang-barang konsumsi oleh massa. Hal ini juga didukung oleh pengembangan moda transportasi dan media massa yang dibutuhkan oleh produsen untuk menemukan cara untuk mengatur distribusi barang dan jasa. Selama Revolusi Industri, barang-barang



konsumen masih langka dan produsen dapat menjual hampir semua yang mereka produksi selama konsumen dapat membeli. Karena itu, mereka fokus pada pengembangan produksi dan distribusi dengan dukungan finansial paling sedikit. Ini juga berfokus pada pengembangan ilmu pemasaran pada saat itu, yang berkonsentrasi pada efisiensi biaya distribusi dan pembukaan pasar baru. Dari awal abad ke-20 hingga beberapa tahun setelah Perang Dunia II, persaingan bisnis telah meningkat dan fokus ilmu pemasaran telah bergeser dari fokus produksi ke fokus penjualan. Komunikasi, periklanan, dan sains merek mulai menjadi penting karena perusahaan mendapatkan banyak barang di pasar yang semakin ramai.

### **1.2.2 Strategi Penjualan**

Kata strategi dalam Kamus Besar Bahasa Indonesia (KBBI) adalah rencana yang cermat dalam mengembangkan program yang sejalan dengan tujuan dan sasaran tertentu. Strategi penjualan adalah cara untuk memperoleh pendapatan dari penjualan (sales) yang disusun berdasarkan kekuatan, keunggulan, yang mendukung perusahaan atau seseorang.

Kemudian diubah menjadi rencana yang dapat ditindaklanjuti untuk menemukan pelanggan baru atau mengembangkan pelanggan yang sudah ada. Sederhananya, strategi penjualan adalah kegiatan yang direncanakan yang bertujuan untuk meningkatkan volume penjualan produk. Baik itu barang atau jasa.

#### **1. Target Pasar**

Hal pertama dalam menentukan strategi penjualan adalah Anda harus mengetahui target pasar untuk produk yang Anda tawarkan. Target pasar Anda adalah kumpulan orang yang ingin Anda fokuskan dalam promosi atau cara lain untuk meningkatkan volume penjualan Anda.



Contoh berikut ini akan memudahkan kita untuk memahami target pasar. Jika Anda seorang penjual yang menjual unit apartemen dengan harga terendah satu miliar rupiah. Jika Anda menawarkan unit apartemen kepada seseorang yang baru memulai karir profesionalnya, maka unit tersebut akan dijual dengan harga yang sangat murah. Lain halnya jika Anda menawarkan unit kepada CEO suatu perusahaan atau perusahaan, itu harus dijual lebih banyak.

Dalam dunia penjualan dan pemasaran tidak ada target pasar "semua orang" bahkan untuk barang-barang berharga murah. Semakin banyak Anda tahu tentang target pasar Anda, semakin baik peluang Anda untuk meningkatkan penjualan.

## 2. Jenis produk

Kedua adalah jenis produk, tidak semua produk sejenis memiliki target pasar yang sama. Coba ambil contoh jus buah sebagai produk.

Jus buah yang kita temukan di banyak restoran atau bahkan di jalan berfungsi hanya terbatas pada jenis minuman. Ini berbeda dari produk jus yang fokus pada nilai kesehatannya. Target pasar dan harga pasti akan berubah. Jus ini akan membantu hal-hal seperti tidak ada gula, diet yang efektif, manfaat kesehatan, kandungan gizi, dan vitamin dalam jus. Hal-hal menarik di atas akan sangat menarik bagi orang yang memprioritaskan kesehatan. Rahasia strategi penjualan sebenarnya adalah layanan dan bukan produk jus. Semua orang bisa membuat jus dengan mudah. Tetapi tidak semua orang memiliki waktu untuk membeli berbagai bahan dan mengolahnya menjadi jus dengan manfaat dan rasa yang sama.

### 1.3 Website



### 1.3.1 Pengertian Website

Web adalah kumpulan informasi / koleksi halaman yang biasanya diakses melalui internet. Setiap orang di berbagai tempat dan kapan saja dapat terhubung saat terhubung ke internet secara online di jaringan. Secara teknis, situs web adalah kumpulan halaman, yang tergabung dalam domain atau subdomain tertentu. Situs web di World Wide Web (WWW).

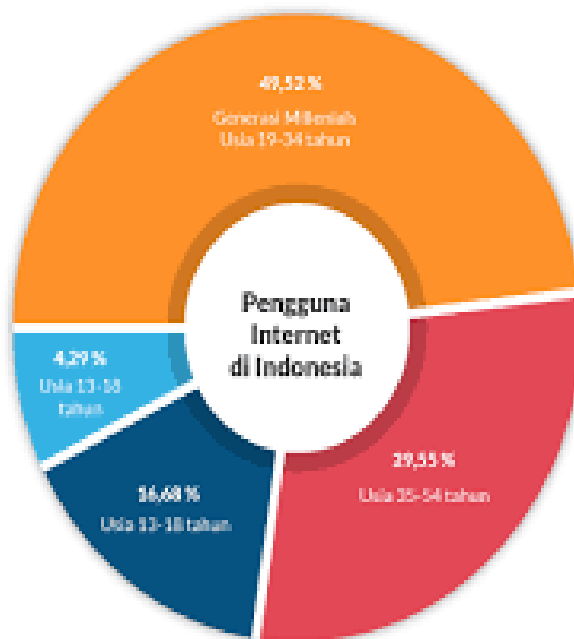
### Memahami Situs Web Menurut Para Ahli

Halaman diakses dan dibaca melalui browser seperti Netscape Navigator, Internet Explorer, Mozilla Firefox, Google Chrome dan aplikasi browser lainnya (pemahaman situs web dari Hakim Lukmanul, 2004)

Situs web adalah kumpulan halaman web yang saling terkait dan file terkait. Web terdiri dari halaman / halaman, dan halaman gabungan yang disebut homepage. Halaman depan berada di posisi teratas, dengan halaman yang ditautkan di bawah. Setiap halaman di beranda bawah disebut halaman anak, yang berisi hyperlink ke halaman lain di web (pemahaman situs web dari Gregory, 2000)

#### **1.3.1 Sejarah Website**

Situs web ini pertama kali ditemukan oleh Sir Timothy John, Tim Berners-Lee. Pada tahun 1991 situs web terhubung ke jaringan. Tujuan situs web dibuat pada waktu itu untuk memfasilitasi pertukaran pertukaran dan memperbarui informasi bagi para peneliti di tempat kerja mereka. Dengan demikian pengertian situs web pada waktu itu masih sebatas bertukar informasi, bukan memahami terminologi situs web.



Situs web menyetujui publik setelah diumumkan dari CERN pada 30 April 1993. CERN menyatakan bahwa situs web dapat digunakan secara gratis oleh semua orang. Pada saat ini, ide situs web telah memasuki ruang publik karena dapat digunakan oleh semua orang di mana saja dan kapan saja.

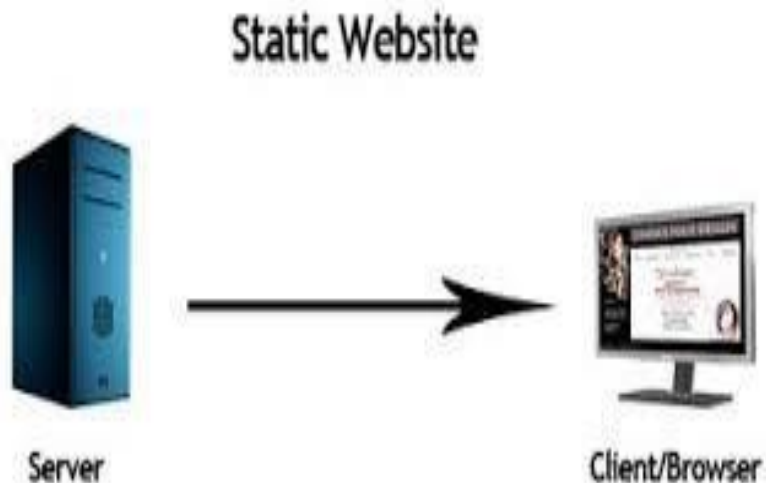
Dimbleweb mencoba menjelaskan jenis dan tipe situs web yang umumnya tersedia.

Secara garis besar, suatu situs web dapat digolongkan menjadi 3 bagian, yaitu:

1. Situs Web Statis

Web Statis adalah web di mana pengguna tidak dapat mengubah situs web secara langsung melalui browser. Jadi web statis ini hanya dapat

diubah secara manual. Sedangkan web dinamis adalah situs web yang kontennya dapat dengan mudah diperbarui dengan mudah



## 2. Situs Web Dinamis

Dynamic Website adalah jenis halaman web yang disusun oleh konten dan tata letak yang kaya akan informasi di dalamnya. Dinamakan situs web dinamis karena kontennya dapat berubah. ... Kemudian, situs web Statis (Situs Web Statis) adalah situs web yang kontennya secara statistik / tidak berubah.



### 3. Situs Web Interaktif

Dynamic Website adalah jenis halaman web yang disusun oleh konten dan tata letak yang kaya akan informasi di dalamnya. Dinamakan situs web dinamis karena kontennya dapat berubah. ... Kemudian, situs web Statis (Situs Web Statis) adalah situs web yang kontennya secara statistik / tidak berubah.

Statistik situs web adalah situs web yang memiliki halaman yang tidak berubah. Ini berarti membuat perubahan pada halaman yang dilakukan secara manual dengan mengubah kode ke dalam struktur situs.

Situs Web Dinamis adalah situs web yang dirancang untuk dimaksudkan untuk pembaruan sesering mungkin. Apa pun yang dapat diakses oleh pengguna secara umum, juga menyediakan halaman backend untuk menginstal konten dari situs web. Contoh umum dari situs dinamis adalah berita web atau portal web yang mencakup berita, fasilitas pemungutan suara, dan sebagainya. Situs web interaktif adalah web yang saat ini sedang booming. Salah satu contoh situs web interaktif adalah blog dan forum. Di situs web para pengguna ini dapat berdiskusi dan bertabrakan argumen

tentang apa yang mereka anggap. Situs web biasa seperti memiliki moderator untuk topik yang disetujui yang dibahas tidak keluar jalur.

Tiga jenis situs di atas adalah jenis situs yang sering dibuat oleh pengembang di Indonesia seperti pembuatan situs web, di mana jenis yang paling diinginkan adalah DINAMIK, karena pemilik situs web dapat membuat perubahan konten secara mandiri.

### **1.3.2 Perkembangan Web**

Sejak web ditemukan pada tahun 1990, Web menjadi semakin populer dan layanan internet dengan pengguna terbanyak. Sebagian besar pengguna internet bahkan berpikir bahwa Web adalah satu-satunya layanan di internet. Teknologi web terus bergulir dari era web 1.0, web 2.0 dan tren menuju web 3.0. Sayangnya, karakteristik antara versi web 1.0, hingga 3.0 tidak disetujui oleh para ahli. Namun, bukan berarti kita tidak tahu polanya. Berikut ini adalah pola dari setiap versi web sesuai dengan sumber rock yang saya baca.

#### **1. Web 1.0**





Web 1.0 dimulai di era web tahun 90-an. Ciri utama era ini adalah informasi statistik.

#### Karakteristik web 1.0

Situs web umumnya statis yang berubah atau tidak berubah sama sekali.

Situs web publik tidak interaktif.

Teknologi yang digunakan adalah teknologi tertutup.

Frame desain web menggunakan frame.

Font kombinasi warna dan teks terlihat norak karena hanya terbatas pada 16 warna dan 6 jenis font.

Informasi yang tersedia umumnya berupa teks dan gambar berita.

#### Teknologi web 1.0

HTML dasar, CSS dasar, dan Javascript.

Aplikasi Flash dan Java.

Browser populer adalah Netscape dan Internet Explorer.

Koneksi internet masih dihubungi hingga maksimal 56 kbps.



## 2. Web 2.0

Web 2.0 dimulai pada era 2005 hingga sekarang, era unik adiri ini adalah konten yang dibuat oleh pengguna dan jejaring sosial.

Karakteristik Web 2.0

Situs web AJAX bersifat dinamis dan interaktif dengan teknologi.

Pengguna tidak hanya sebagai konsumen tetapi juga sebagai produsen informasi dalam istilah populer yang dibuat oleh pengguna konten (misalnya youtube atau wikipedia).

Teknologi yang digunakan umumnya open source.

Desain web menggunakan tata letak CSS.

Informasi terdiri dari teks, gambar, audio, video dan animasi.

Web sebagai media komunikasi dan kolaborasi.

Jejaring sosial mendominasi di era ini.

Munculnya startup.

Munculnya jenis situs web seperti microbloging, berbagi foto dan video, bookmark online, blog dan jejaring sosial.

Teknologi di Web 2.0

Pustaka Ajax dan jQuery dan Javascript mendominasi.

Tingkatkan penggunaan HTML5 dan kurangi penggunaan Flash.

Web Seluler dan desain web Responsif.

- XML dan JSON untuk transfer data.

API · Web dan Mashup (Mashup adalah istilah yang berisi layanan bersama, misalnya menampilkan tweet di peta Google).

Browser populer adalah Chrome, Safari, dan Firefox.

- · Koneksi internet menggunakan broadband (> 1MB / s).

Android dan iOS menyetujui pengguna web seluler.

### 3. Web 3.0

Web 3.0 adalah generasi berikutnya dari teknologi web. Fitur Web 3.0 adalah akses seluler, web semantik, dan personalisasi. Beberapa situs web yang dianggap berasal dari web 3.0 adalah Instagram. Instagram

disebut sebagai permulaan generasi web 3.0 karena layanan ini hanya tersedia melalui perangkat seluler.

### Karakteristik Web 3.0

Integrasi web ke dalam produk-produk Rumah seperti TV, Kulkas atau Windows, smart atau home entertainment. Contoh dari produk ini adalah Smart Windows dari Samsung atau Google Nexus Q.

Web Semantik, Contoh penggunaan web semantik adalah grafik Pengetahuan di Google Search.

Memproses bahasa alami. Perintah suara. ada beberapa contohnya adalah Siri, Voice Action Android atau Voice search di Google.

Layanan berbasis lokasi dan personifikasi informasi. Web tidak lagi menyediakan informasi, tetapi solusi. Contoh layanan ini adalah Layar dan Google Now.

Komputer (dan web) yang dapat digunakan, contoh produk ini adalah Google glass.

### Teknologi Web 3.0

Untuk teknologi yang mendominasi web 3.0 adalah sebagai berikut:

Semantic Web (Contoh bagan pengetahuan).

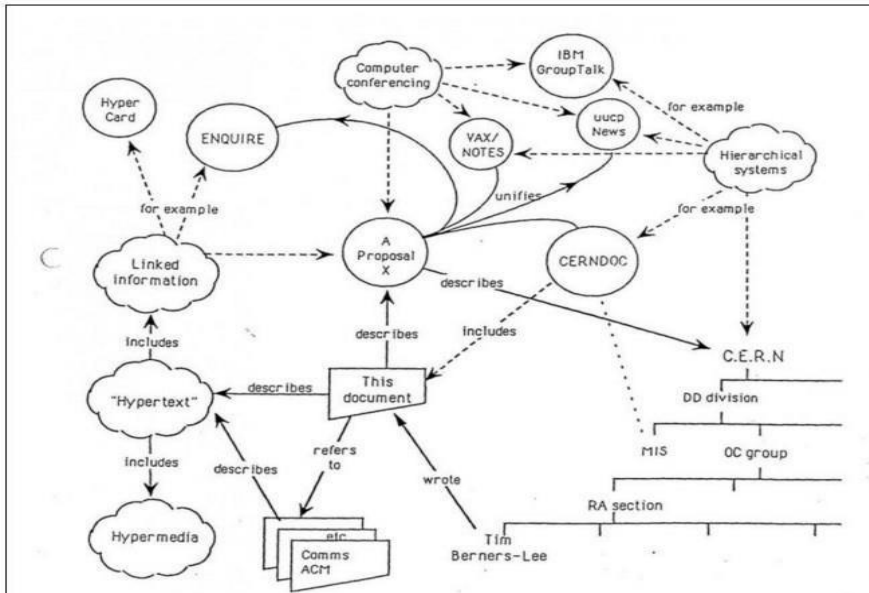
Kecerdasan Buatan (Siri).

Komunikasi & kolaborasi waktu nyata (Google docs, Google Wave).

Augment Reality (Layar, aplikasi di android).

Informasi yang dihasilkan komputer (Contoh Wolfram Alpha).

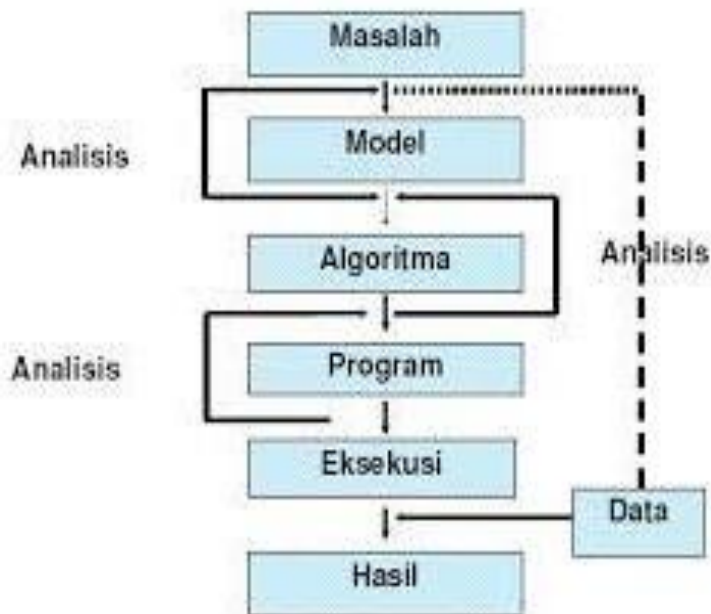
HTML5 dan teknologi Cloud.



## 1.4 Algoritma Metode Serakah

### 1.3.2 Metode Serakah

Algoritma serakah adalah algoritma yang mirip dengan pemrograman dinamis, bedanya jawabannya tidak perlu diketahui dalam setiap tahap; dan menggunakan pilihan “serakah” apa yang dilihat terbaik pada saat itu.

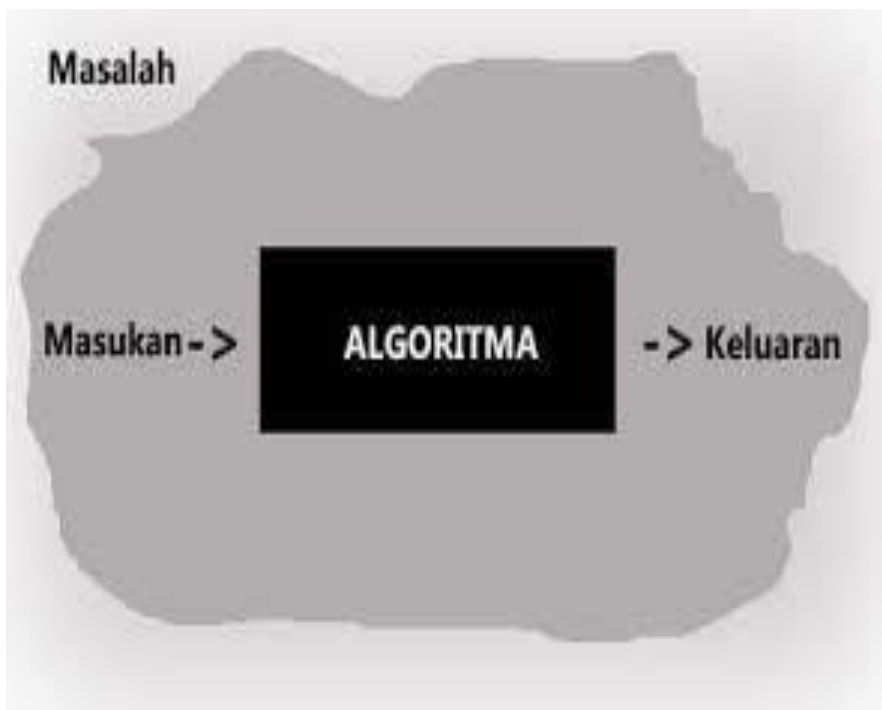


Algoritma serakah adalah algoritma yang mengikuti heuristik pemecahan masalah dalam membuat pilihan optimal pada setiap langkah dengan tujuan menemukan global optimal. Dalam banyak masalah, strategi serakah biasanya tidak menghasilkan solusi optimal, tetapi heuristik serakah dapat menghasilkan solusi optimal lokal yang membutuhkan solusi optimal global dalam jumlah waktu yang wajar. contohnya, strategi serakah untuk masalah salesman keliling (yang memiliki kompleksitas komputasi tinggi) adalah heuristik berikut: "Pada setiap langkah, kunjungi kota terdekat yang belum dikunjungi." Heuristik ini tidak setuju untuk menemukan solusi terbaik, tetapi diakhiri dengan menyimpulkan langkah yang masuk akal; Temukan solusi optimal untuk masalah yang sangat rumit. Dalam optimisasi matematis, suatu algoritma yang memecahkan masalah kombinatorial yang memiliki sifat matematika, dan memberikan estimasi faktor konstan untuk masalah optimisasi dengan struktur submodular.

Secara umum, algoritma serakah memiliki lima komponen:

1. Kumpulan kandidat, dari mana solusinya dibuat
2. Fungsi seleksi, yang memilih kandidat terbaik untuk ditambahkan ke solusi
3. Fungsi kelayakan, yang digunakan untuk menentukan apakah seorang kandidat dapat digunakan untuk berkontribusi pada solusi
4. Fungsi obyektif, yang memberi nilai pada solusi, atau solusi parsial, dan
5. Fungsi Solusi, yang akan ditampilkan setiap kali kami menemukan solusi yang lengkap

Algoritma memberikan solusi yang baik untuk beberapa masalah matematika, tetapi tidak untuk yang lain. Sebagian besar masalah di mana mereka bekerja akan memiliki dua properti:



### 1.4.1 Properti pilihan serakah

Kita dapat membuat pilihan apa pun yang diselesaikan sekarang dan kemudian menyelesaikan submasalah yang muncul kemudian. Pilihan yang dibuat oleh algoritma serakah mungkin bergantung pada pilihan yang dibuat, tetapi tidak pada pilihan di masa depan atau semua solusi untuk submasalah. Iteratif membuat satu pilihan serakah demi satu, mengurangi setiap masalah ke tingkat yang lebih kecil. Dengan kata lain, algoritma serakah tidak pernah bisa memikirkan kembali pilihannya. Ini adalah perbedaan utama dari pemrograman dinamis, yang lengkap dan dijamin untuk menemukan solusi. Setelah setiap instalasi, pemrograman dinamis membuat keputusan berdasarkan semua keputusan yang dibuat dalam instalasi sebelum. Algoritma memiliki sejarah panjang dalam ilmu komputer kombinatorial dan teoritis. Heuristik rakus Berhasil menghasilkan hasil pada masalah optimal, [4] dan pertanyaan alami adalah:

- Untuk masalah apa algoritma serakah bekerja secara optimal?
- Untuk masalah apa algoritma serakah meminta solusi optimal?
- Untuk masalah apa algoritma serakah yang optimal tidak menghasilkan solusi yang optimal?

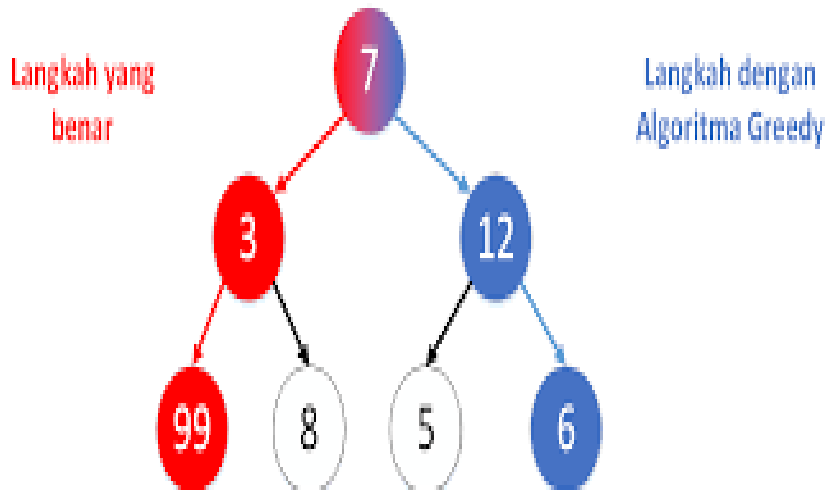
Ada banyak literatur yang menjawab pertanyaan ini untuk kelas masalah umum, seperti matroid, serta untuk masalah khusus, seperti set cover,nya, dan dapat mempertimbangkan kembali jalur algoritmik.

Sebagian besar algoritma serakah (tetapi tidak selalu) gagal menemukan solusi global optimal yang mereka lakukan tidak dapat dilakukan pada semua data. Mereka dapat membuat komitmen untuk pilihan-pilihan tertentu sejak dini yang mencegah mereka menemukan solusi terbaik nanti. Sebagai



contoh, semua algoritma pewarnaan serakah yang dikenal untuk masalah pewarnaan grafik dan semua masalah NP lengkap lainnya tidak menemukan solusi optimal. Meskipun demikian, mereka berguna karena mereka berpikir cepat dan sering memberikan perkiraan yang optimal. Jika algoritma serakah dapat dibuktikan menghasilkan global yang optimal untuk kelas masalah yang diberikan, biasanya merupakan metode yang disukai karena lebih cepat daripada metode optimasi seperti pemrograman dinamis. Contoh dari algoritma serakah ini adalah algoritma Kruskal dan algoritma Prim untuk menemukan pohon rentang minimum, dan algoritma untuk menemukan pohon Huffman yang optimal.

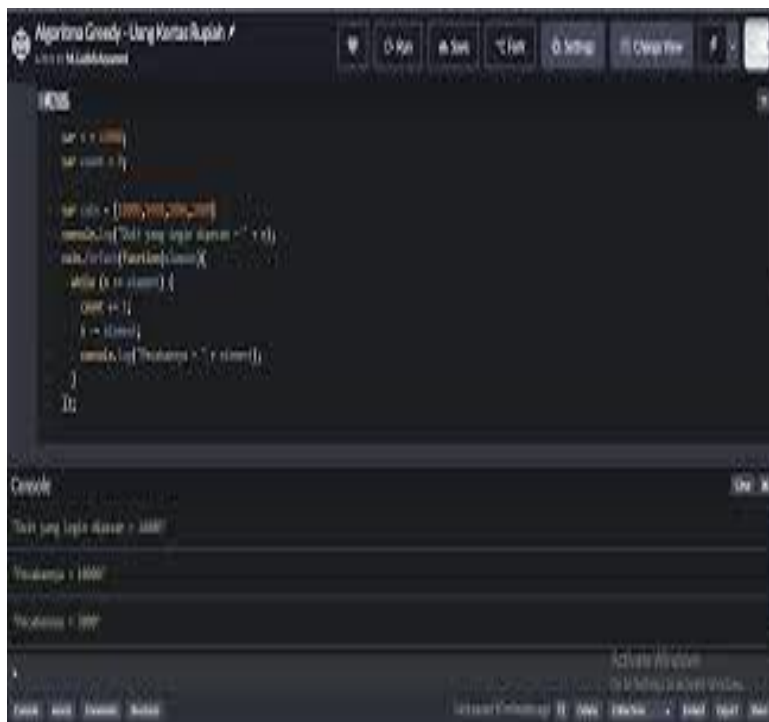
## Pencarian Nilai Terbesar



Algoritma serakah juga muncul dalam perutean jaringan. Menggunakan rute serakah, pesan diteruskan ke jalur tetangga yang "paling dekat" ke tujuan. Lokasi node (dan "proximity" yang ditentukan) dapat ditentukan oleh lokasi fisiknya, seperti pada rute geografis yang digunakan

oleh jaringan ad hoc. Lokasi mungkin juga merupakan konstruksi yang dibuat seperti dalam routing dunia kecil dan tabel hash yang didukung.

Ada beberapa contoh gambar dari algoritma serakah, bias dilihat pada gambar diawah ini



```
Algoritma Greedy - Menggunakan Rujukan /  
11/11/2020 10:00:00  
#include <iostream>  
using namespace std;  
int main() {  
    int n;  
    cin >> n;  
    int *a = new int[n];  
    for (int i = 0; i < n; i++) {  
        a[i] = rand() % 100;  
    }  
    int sum = 0;  
    for (int i = 0; i < n; i++) {  
        sum += a[i];  
    }  
    cout << sum << endl;  
    return 0;  
}
```

Console

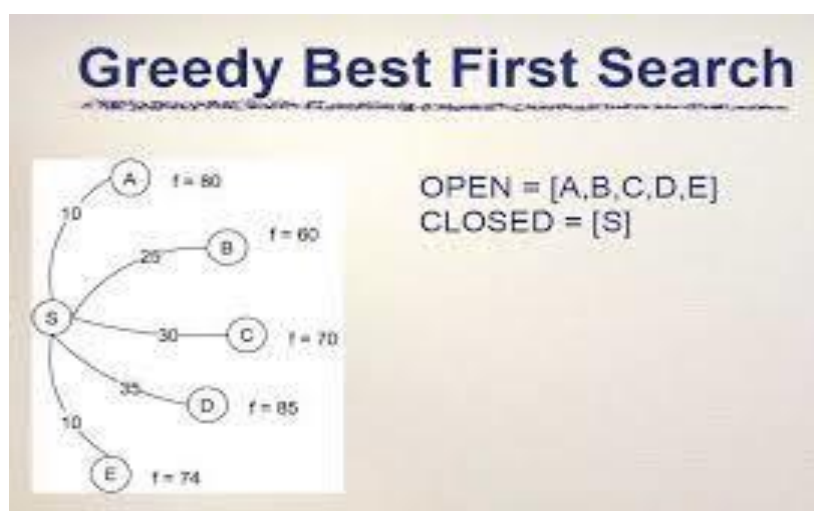
Task yang ingin dijawab: 1000

Hasilnya: 1000

Hasilnya: 1000

Activate Windows  
Go to Settings to activate Windows.

File Edit View Help



- Masalah pemilihan kegiatan adalah masalah dari kelas masalah ini, disepakati untuk memilih jumlah maksimum kegiatan yang tidak saling berbenturan.
- Dalam game komputer Macintosh, Quest Crystal, mengumpulkan kristal koleksi, dengan cara yang mirip dengan masalah salesman keliling. Gim ini memiliki mode demo, di mana ia menggunakan algoritma serakah untuk setiap kristal. Mode demo tidak disetujui, mode demo sering berakhir dengan cepat.
- Pengejaran yang cocok adalah contoh algoritma serakah yang diterapkan pada estimasi sinyal.
- Algoritma serakah menemukan solusi optimal untuk masalah Malfatti dalam menemukan tiga lingkaran terputus-putus dalam segitiga tertentu yang memaksimalkan luas total lingkaran; Terkait dengan algoritma serakah yang sama adalah optimal untuk lingkaran yang terbagi.
- Algoritma ini digunakan untuk membangun pohon Huffman selama pengkodean Huffman di mana ia menemukan solusi optimal.
- Dalam pembelajaran pohon keputusan, algoritma serakah biasanya digunakan, tetapi tidak disetujui untuk menemukan solusi optimal.
- Salah satu algoritma yang populer adalah algoritma ID3 untuk konstruksi pohon keputusan.
- Algoritma Dijkstra dan algoritma pencarian terkait A \* adalah algoritma serakah yang benar-benar optimal untuk pencarian grafik dan pencarian jalur terpendek.

- Pencarian bersyarat opsional, membutuhkan "heuristik yang dapat diterima" yang tidak akan melebihi biaya jalur.
- Algoritma Kruskal dan algoritma Prim adalah algoritma serakah untuk membuat pohon rentang minimum dari grafik terhubung yang diberikan. Mereka selalu menemukan solusi optimal, yang mungkin tidak unik secara umum.

## BAB 2

---

### TEORI

---

#### 2.1 Sejarah CI

*CodeIgniter* adalah aplikasi sumber terbuka yang berupa kerangka kerja *PHP* dengan Model, View, Controller. Untuk membangun website dinamis dengan menggunakan PHP. *CodeIgniter* memudahkan pengembangan web untuk membuat aplikasi web dengan cepat mudah dibandingkan dengan membuatnya dari awal. *CodeIgniter* ini dirilis pertama kali pada 28 Februari 2006. Versi stabil terakhir adalah versi 3.1.11

##### 2.1.1 Kerangka Kerja

Kerangka kerja secara sederhana dapat diartikan kumpulan dari fungsi-fungsi atau prosedur-prosedur dan *class-class* untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang *programer*, tanpa harus membuat fungsi atau *class* dari awal.

Ada beberapa alasan mengapa menggunakan Kerangka kerja:

- Mempercepat dan mempermudah pembangunan sebuah aplikasi web.
- Relatif memudahkan dalam proses *maintenance* karena sudah ada pola

tertentu dalam sebuah kerangka kerja (dengan  
syarat *programmer* mengikuti pola standar yang ada

Umumnya kerangka kerja menyediakan fasilitas-fasilitas yang umum dipakai sehingga kita tidak perlu membangun dari awal (misalnya *validasi, ORM, pagination, multiple database, scaffolding, pengaturan session, error handling*, dll).

- Lebih bebas dalam pengembangan jika dibandingkan *Cms*.

### **2.1.2 Design Patern : MVC ( Model, View, Controller )**

*Model View Controller* merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman *Small Talk*, *MVC* memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, *user interface*, dan bagian yang menjadi kontrol aplikasi. Terdapat 3 jenis komponen yang membangun suatu *MVC pattern* dalam suatu aplikasi yaitu:

1. *View*, merupakan bagian yang menangani *presentation logic*. Pada suatu aplikasi web bagian ini biasanya berupa *file template HTML*, yang diatur oleh *controller*. *View* berfungsi untuk menerima dan merepresentasikan data kepada *user*. Bagian ini tidak memiliki akses langsung terhadap bagian model.
2. Model, biasanya berhubungan langsung dengan *database* untuk memanipulasi data (*insert, update, delete, search*), menangani *validasi* dari bagian *controller*, tetapi tidak dapat berhubungan langsung dengan bagian *view*.
3. *Controller*, merupakan bagian yang mengatur hubungan antara bagian model dan bagian *view*, *controller* berfungsi untuk menerima *request* dan data dari *user* kemudian menentukan apa yang akan diproses oleh aplikasi.

Dengan menggunakan prinsip *MVC* suatu aplikasi dapat dikembangkan sesuai dengan kemampuan *developernya*, yaitu programmer yang menangani bagian model dan *controller*, sedangkan *designer* yang menangani bagian *view*, sehingga penggunaan arsitektur *MVC* dapat meningkatkan *maintanability* dan organisasi kode. Walaupun demikian dibutuhkan komunikasi yang baik antara programmer dan *designer* dalam menangani variabel-variabel yang akan ditampilkan.

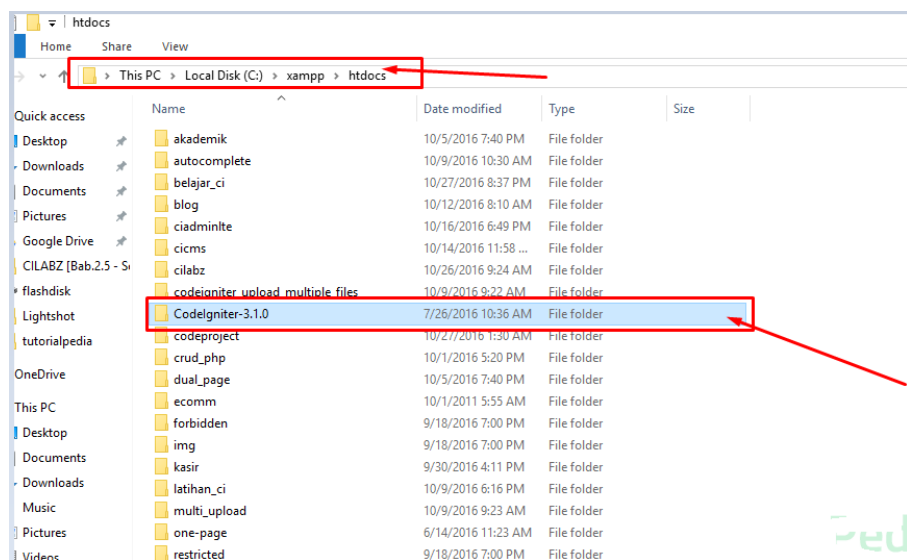
Ada beberapa kelebihan *CodeIgniter (CI)* dibandingkan dengan Kerangka kerja *PHP* lain,

- Performa sangat cepat: salah satu alasan tidak menggunakan kerangka kerja adalah karena eksekusinya yang lebih lambat daripada *PHP from the scratch*, tapi *Codeigniter* sangat cepat bahkan mungkin bisa dibilang *codeigniter* merupakan kerangka kerja yang paling cepat dibanding kerangka kerja yang lain.
- *Konfigurasi* yang sangat minim (*nearly zero configuration*): tentu saja untuk menyesuaikan dengan *database* dan keleluasaan *routing* tetap diizinkan melakukan *konfigurasi* dengan mengubah beberapa *file konfigurasi* seperti *database.php* atau *autoload.php*, tetapi untuk menggunakan *codeigniter* dengan *setting standard*, anda hanya perlu mengubah sedikit saja *file* pada *folder config*.
- Banyak komunitas: dengan banyaknya komunitas *CI* ini, memudahkan kita untuk berinteraksi dengan yang lain, baik itu bertanya atau teknologi terbaru.
- Dokumentasi yang sangat lengkap: Setiap paket *instalasi codeigniter* sudah disertai *user guide* yang sangat bagus dan lengkap untuk dijadikan permulaan, bahasanya pun mudah dipahami.



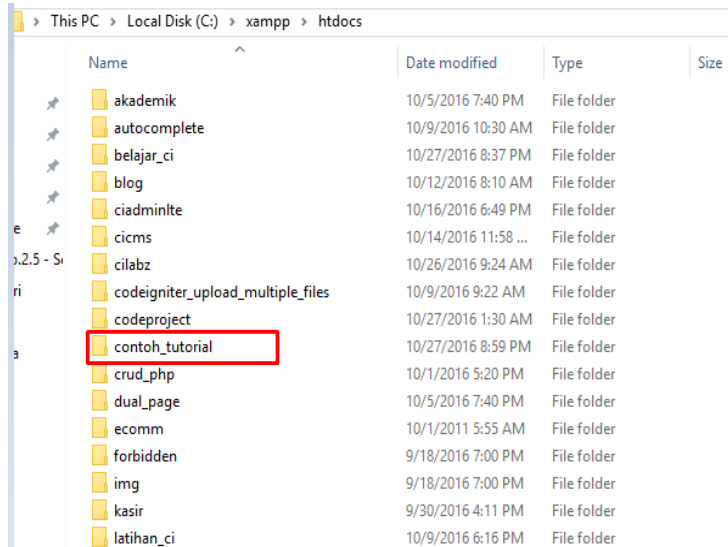
## 2.2 Instalasi Codeigniter

- *Download Package codeigniter* pada situ resminya. *Download Codeigniter*
- Saat ini versi stabil dari *codeigniter* adalah 3.1.1.
- Setelah *di download ekstrak file codeigniter* pada direktori *C:\xampp\htdocs*



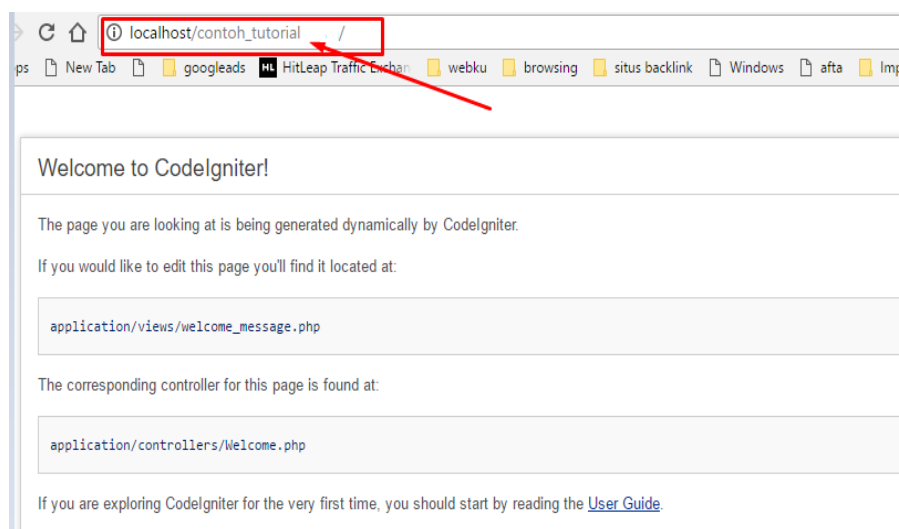
Gambar 2.1

- lalu ubah folder **Codeigniter** menjadi sesuai keinginan kita. disini saya akan ubah folder *codeigniter* tersebut menjadi **contoh\_tutorial**



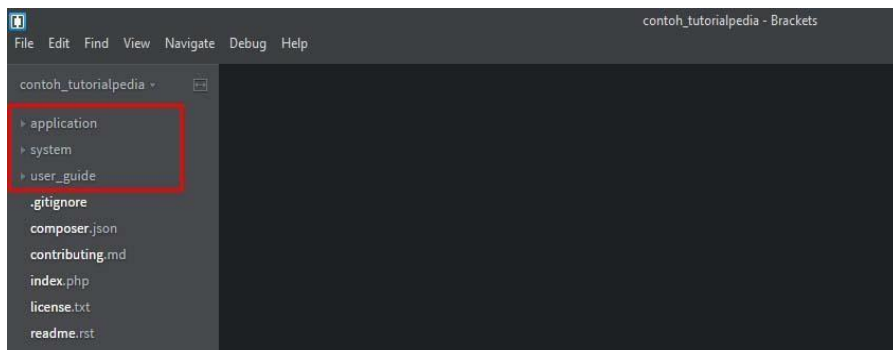
Gambar 2.2

- Setelah itu, sekarang kita coba akses folder tersebut melalui *browser*.
- jangan lupa untuk menyalakan *XAMPP* terlebih dahulu.
- lalu akses dengan url `http://localhost/nama_folder_project` pada tutorial kali ini saya beri nama dengan `contoh_tutorial`



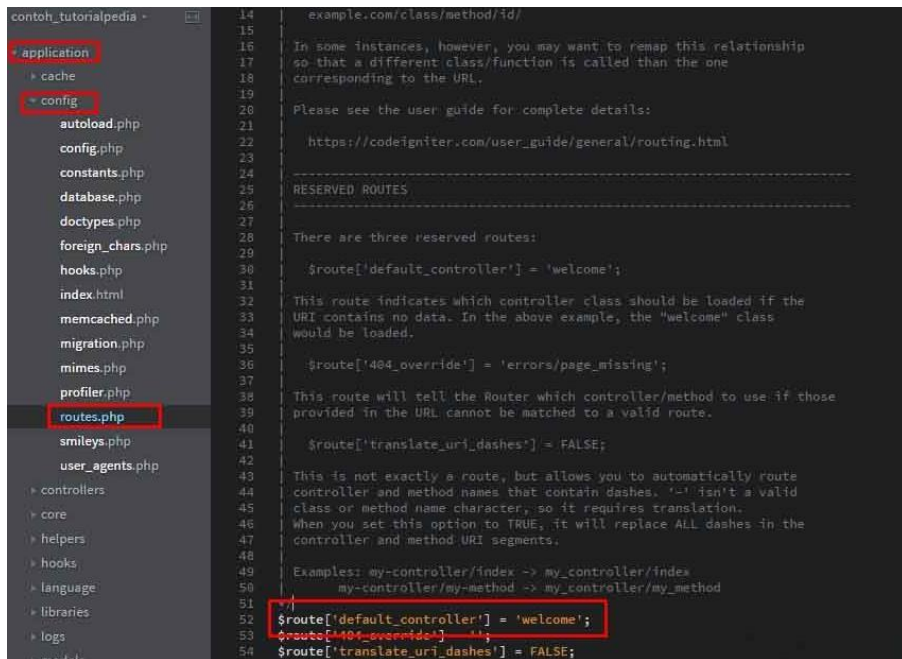
Gambar 2.3

- Jika sudah tampil halaman **Welcome Codeigniter**, itu artinya anda telah berhasil melakukan *instalasi codeigniter*.
- Penjelasan Tentang Struktur Folder *Codeigniter*
- Didalam folder *codeigniter* terdapat 3 folder utama yaitu *Application, System, User Guide*. Folder *application* berguna sebagai tempat untuk mendvelop aplikasi kita nanti, Folder *system* berisi library atau kumpulan fungsi-fungsi dasar CI, sedangkan *User Guide* berisi sebagai buku panduan atau dokumentasi dari *codeigniter*.



Gambar 2.4

Lalu kita fokus pada folder **application/config** dan semua berawal dari file **routes.php**. Apabila kita buka file tersebut, maka kita akan menemukan **default\_controller** yang telah terisi dengan nama **“welcome”**, yang mana **default\_controller** ini berfungsi ketika user tidak melakukan atau memasukkan apapun pada URL browser, maka secara otomatis mencari controller dengan nama **“welcome”**. Seperti dibawah ini :



Gambar 2.5

**Controller Welcome** itu sendiri bisa anda temukan pada folder **application/controllers**. Pada file **welcome\_controller** tersebut terdapat **fungsi index** yang gunanya **meload** bagian **view** atau memanggil halaman **view** dengan nama **welcome\_messages** sebagai berikut.



Gambar 2.6

Sekarang kita lihat pada bagian **views** yang terletak pada **application/views**. Pada folder **view** tersebut terdapat file dengan nama **welcome\_message**. Pada bagian inilah semua **html** dan **css** akan diletakkan yang nantinya akan di tampilkan ke pengunjung.

## 2.3 XAMPP

XAMPP adalah sebuah paket perangkat lunak (*software*) komputer yang sistem penamaannya diambil dari *akronim* kata *Apache*, *MySQL* (dulu) / *MariaDB* (sekarang), *PHP*, dan *Perl*. Sementara imbuhan huruf “X” yang terdapat pada awal kata berasal dari istilah *cross platform* sebagai simbol bahwa aplikasi ini bisa dijalankan di empat sistem operasi berbeda, seperti *OS Linux*, *OS Windows*, *Mac OS*, dan juga *Solaris*.

Sejarah mencatat, *software* XAMPP pertama kali dikembangkan oleh tim proyek bernama *Apache Friends* dan sampai saat ini sudah masuk dalam rilis versi **7.3.9** yang bisa didapatkan secara gratis dengan label *GNU (General Public License)*.

Jika dijabarkan secara gamblang, masing-masing huruf yang ada di dalam nama XAMPP memiliki arti sebagai berikut ini:

*X = Cross Platform*

Merupakan kode penanda untuk *software cross platform* atau yang bisa berjalan di banyak sistem operasi.

*A = Apache*

*Apache* adalah aplikasi web server yang bersifat gratis dan bisa dikembangkan oleh banyak *orang (open source)*.

*M = MySQL / MariaDB*

*MySQL* atau *MariaDB* merupakan aplikasi database server yang dikembangkan oleh orang yang sama. *MySQL* berperan dalam mengolah,

mengedit, dan menghapus daftar melalui database.

*P = PHP*

Huruf “P” yang pertama dari akronim kata XAMPP adalah inisial untuk menunjukkan eksistensi bahasa pemrograman PHP. Bahasa pemrograman ini biasanya digunakan untuk membuat website dinamis, contohnya dalam website berbasis *CMS WordPress*.

*P = Perl*

Sementara itu, untuk huruf P selanjutnya merupakan singkatan dari bahasa *pemrograman Perl* yang kerap digunakan untuk memenuhi berbagai macam kebutuhan. Perl ini bisa berjalan di dalam banyak sistem operasi sehingga sangat fleksibel dan banyak digunakan.

### Fungsi XAMPP

Program aplikasi XAMPP berfungsi sebagai server lokal untuk mengampu berbagai jenis data website yang sedang dalam proses pengembangan. Dalam prakteknya, XAMPP bisa digunakan untuk menguji kinerja fitur ataupun menampilkan konten yang ada didalam website kepada orang lain tanpa harus terkoneksi dengan internet, atau istilahnya *website offline*. XAMPP bekerja secara *offline* layaknya *web hosting* biasa namun tidak bisa diakses oleh banyak orang. Maka dari itu, XAMPP biasanya banyak digunakan oleh para mahasiswa maupun pelajar untuk melihat hasil desain website sebelum akhirnya dibuat online menggunakan *web hosting* yang biasa dijual dipasaran.

#### 2.3.1 Instalasi XAMPP

1. Anda harus menyiapkan file XAMPP terlebih dahulu dengan cara

mengunduh

lewat

*link*

berikut



ini <http://www.apachefriends.org/en/index.html>.

File ini

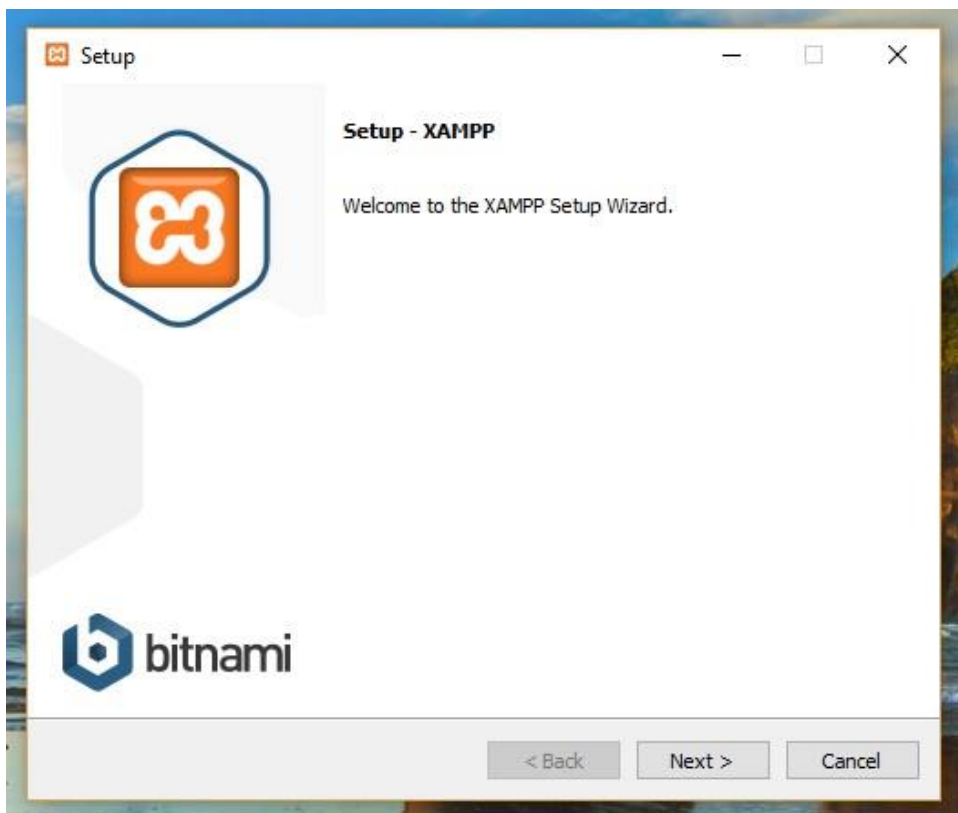
bersifat gratis.



Halaman download aplikasi XAMPP

Gambar 2.7

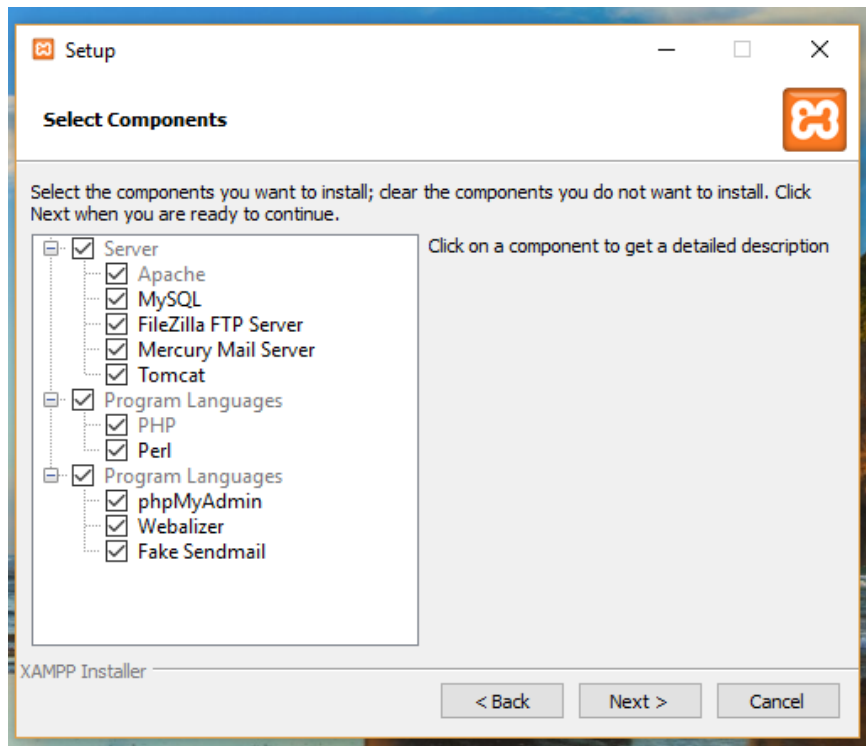
1. Setelah file berhasil didownload dengan sempurna langkah selanjutnya adalah menyiapkan *space/ruang harddisk* di laptop atau komputer dengan minimal masih menyisakan *ROM* sebesar 1GB guna menampung gambar, teks, video, dan dokumen website.
2. Jalankan file *xampp-windows-x64-7.3.9-0-VC15-installer* atau versi lainnya yang lebih mutakhir.
3. Akan ada jendela baru yang terbuka sebagai indikasi dimulainya proses *instalasi*. Pada proses ini Anda akan diminta untuk memilih bahasa Indonesia atau *English*, pilih saja yang bahasa Indonesia kemudian klik *next*



*langkah pertama install xampp*

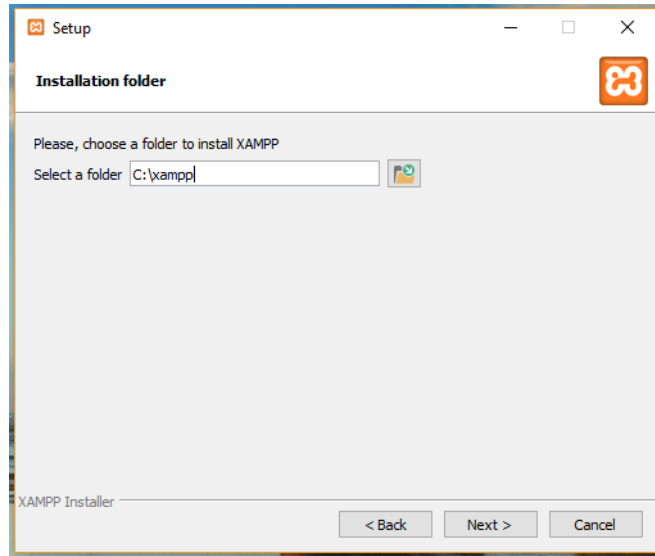
Gambar 2.8

4. Setelah itu, Anda akan diminta untuk memilih komponen yang akan digunakan. Standar untuk *server web* berbasis *CMS WordPress* menggunakan *MySQL*, *FileZilla FTP server*, dan *Apache*, sedangkan dari bahasa pemrograman bisa menceklist pilihan *phpMyAdmin*, *PHP*, dan *Perl*. Kemudian klik *next*.



Gambar 2.9

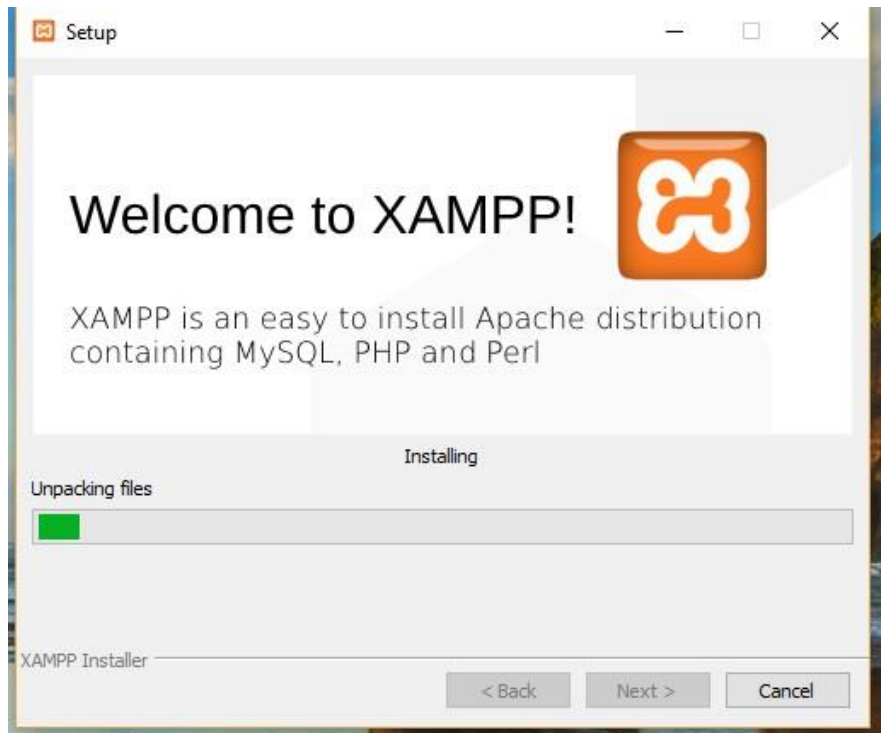
5. Pada tahap ini kita diminta untuk memilih lokasi *file XAMPP* yang akan diinstal, gunakan saja pengatutan *default* lalu klik *next/install*.



*menentukan lokasi file disimpan*

Gambar 2.10

6. Tunggu beberapa menit sampai proses *instalasi XAMPP* selesai yang ditandai dengan bar progress seluruhnya berubah menjadi warna hijau.



*proses install XAMPP*

Gambar 2.9

7. Anda sudah bisa menjalankan *program XAMPP* di laptop atau *PC Windows* untuk membuat *website* secara *offline* menggunakan *localhost*.

### **2.3.2 Bagian Penting yang ada pada XAMPP**

#### *1. Htdocs*

*Htdocs* adalah nama sebuah *folder* bagian dari *XAMPP* yang berfungsi untuk menyimpan berbagai *file* dan dokumen yang akan ditampilkan kedalam *website*. Kapasitas penyimpanan di dalam menu *htdocs* sendiri bergantung kepada harddisk yang terpasang di perangkat komputer.

## 2. Control Panel

Control Panel memberikan kita akses lebih leluasa dalam mengatur *database*, mengunggah *file*, atau melakukan *setting* lebih detail terkait website. Dari dalam *Control Panel* kita juga bisa mengatur *stop* dan *star* aplikasi *XAMPP* agar tidak memberatkan kinerja komputer

## 3. PhpMyAdmin

Sama seperti fungsi *PhpMyAdmin* pada *web hosting* sesungguhnya, peran *PhpMyAdmin* dalam aplikasi *XAMPP* juga *diplot* sebagai pengatur *konfigurasi MySQL*. Untuk membuka *phpMyAdmin* di *XAMPP* caranya Anda bisa langsung mengetikkan URL *http://localhost/phpMyAdmin*.

## 2.4 Bootstrap

- *Bootstrap* adalah *open-source framework* “kerangka kerja pemrograman”, *front-end “library”* yang bebas “gratis” untuk merancang situs web dan aplikasi web. *Framework* ini berisi template desain berbasis *HTML* dan *CSS* untuk tipografi, formulir, tombol, navigasi dan komponen antarmuka lainnya, serta juga ekstensi opsional *JavaScript*. Tidak seperti kebanyakan *framework* web lainnya, *framework* ini hanya fokus pada pengembangan *front-end* saja.
- *Bootstrap* adalah *proyek* yang paling banyak dibintangi kedua di *GitHub*, dengan lebih dari 126.000 bintang.
- *Bootstrap*, awalnya bernama *Twitter Blueprint*, dikembangkan

oleh Nirav Panchal dan Jacob Thornton di *Twitter* sebagai

kerangka kerja untuk mendorong konsistensi di perangkat internal yang sesuai. Sebelum *Bootstrap*, berbagai *library* “kumpulan kode-kode pemrograman” sudah banyak digunakan untuk pengembangan antarmuka, yang menyebabkan ketidak-konsistenan dan beban pemeliharaan yang berat. Menurut Mark Otto, pengembang Twitter:

- "Sebuah grup developer super kecil dan juga aku telah bersama-sama untuk mendesain dan membangun perangkat internal baru dan melihat kesempatan untuk melakukan hal yang lebih banyak lagi. Melalui proses itu, kami melihat bahwa kami sedang membangun sesuatu yang lebih substansial dari perangkat internal lainnya. Beberapa bulan kemudian, kami menghasilkan sebuah versi paling awal dari Bootstrap, sebagai sebuah cara untuk berbagi pola desain yang lazim dan aset-aset perusahaan."
- Setelah beberapa bulan pengembangan oleh sebuah kelompok kecil, banyak pengembang di Twitter mulai memberikan kontribusi untuk proyek ini sebagai bagian dari Hack Week, sebuah pekan ala hackathon untuk tim pengembangan Twitter. Nama itu diganti dari Twitter Blueprint menjadi Bootstrap, dan dirilis sebagai proyek open-source pada 19 agustus 2011.<sup>[3]</sup> Hal ini terus dipertahankan oleh Mark Otto, Jacob Thornton, dan sekelompok kecil pengembang inti, serta sejumlah besar komunitas dari para kontributor.<sup>[4]</sup>
- Pada 31 Januari 2012, Bootstrap 2 dirilis, yang menambahkan system layout dua belas kolom grid yang responsif, dukungan



built-in untuk Glyphicons, beberapa komponen baru, serta perubahan dari beberapa komponen yang sudah ada.<sup>[5]</sup>

- Pada 19 Agustus 2013, Bootstrap 3 dirilis, yang didesain ulang komponen untuk menggunakan desain flat, dan pendekatan 'mobile-first'.<sup>[6]</sup>
- Pada 29 Oktober 2014, Mark Otto mengumumkan bahwa Bootstrap 4 sedang dalam pengembangan.<sup>[7]</sup> Versi alpha pertama dari Bootstrap 4 dirilis pada 19 Agustus 2015.<sup>[8]</sup> Versi beta pertama dirilis pada 10 Agustus 2017.<sup>[9]</sup> Versi stabil Bootstrap 4 dirilis pada 18 Januari 2018 yang pada dasarnya mengimprovisasi style cetak, utilitas 'border' dan menyediakan lebih banyak kontrol atas flexbox.

## 2.5 MySQL

*MySQL* adalah sebuah perangkat lunak sistem manajemen basis data SQL database management system atau *DBMS* yang multialur, multi pengguna, dengan sekitar 6 juta instalasi di seluruh dunia. *MySQL AB* membuat *MySQL* tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public License. Tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus di mana penggunaannya tidak cocok dengan penggunaan *GPL*.

Tidak sama dengan proyek-proyek seperti *Apache*, yang mana perangkat lunak dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing-masing, *MySQL* dimiliki dan disponsori oleh sebuah perusahaan komersial *Swedia MySQL AB*, di mana memegang hak cipta hampir atas semua kode sumbernya. Kedua orang Swedia dan satu

orang Finlandia yang mendirikan MySQL AB adalah David Axmark, Allan Larsson, dan Michael "Monty" Widenius.

## **2.6 PHP**

*PHP: Hypertext Preprocessor* adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk memprogram situs web dinamis. PHP dapat digunakan untuk membangun sebuah CMS.

## **2.7 CMS**

Merupakan perangkat lunak yang digunakan untuk menambahkan atau memanipulasi atau mengubah isi dari suatu situs web. Umumnya, sebuah CMS (*Content Management System*) terdiri dari dua elemen:

- aplikasi manajemen isi (*Content Management Application, CMA*)
- aplikasi pengiriman isi (*content delivery application, CDA*)

Elemen CMA digunakan untuk mengelola konten yang mungkin tidak memiliki pengetahuan mengenai HTML untuk mengelola pembuatan, perubahan, dan penghapusan isi dari suatu situs web tanpa perlu memiliki keahlian sebagai seorang *webmaster*. Elemen CDA menggunakan dan menghimpun informasi-informasi yang sebelumnya telah ditambah, dikurangi atau diubah oleh pemilik situs web untuk memperbaharui situs web tersebut. Kemampuan atau fitur dari sebuah sistem CMS berbeda-beda. Walaupun begitu, kebanyakan dari perangkat lunak ini memiliki fitur publikasi berbasis *Web*, manajemen format, kontrol revisi, pembuatan indeks, pencarian, dan pengarsipan.

CMS merupakan situs web yang menerapkan sistem ini berorientasi terhadap konten. Sudah bukan merupakan kendala yang berarti bagi manajemen atau humas suatu perusahaan/institusi/organisasi untuk memperbaharui situs webnya. Dengan hak akses dan otoritas masing-masing, setiap bagian dari perusahaan/institusi/organisasi dapat memberikan kontribusinya kedalam website tanpa prosedur yang sulit.

Pada umumnya sebuah CMS memiliki 2 bagian kategori yaitu bagian *Front-end* dan *Back-end*.

Kecanggihan dan fitur masing-masing CMS bergantung pada CMS yang digunakan. Penggunaan sistem hierarki pengguna yang diterapkan CMS dalam hak aksesnya pun sangat bervariasi sesuai CMS masing-masing. Mulai dari level akses user anggota yang hanya dapat mengirimkan data tertentu berupa komentar, kemudian editor yang dapat mengirimkan suatu artikel/berita (untuk CMS yang menyediakan fasilitas ini), hingga level administrator yang dapat melakukan semua fitur yang ada.

Keberadaan aplikasi gratisan di Internet dan juga komunitas sumber terbuka yang semakin menjamur ikut memberikan andil yang signifikan untuk menjadikan teknologi CMS menjadi murah dari segi harga akan tetapi dengan fitur-fitur yang semakin lengkap dan canggih. Dari segi biaya implementasi pembuatan CMS sangatlah murah apalagi jika dibandingkan dengan fitur-fitur dan kemudahan yang ditawarkan. Hal inilah yang akhirnya ikut mempopulerkan CMS dan akhirnya akan menggantikan semua website konvensional yang ada.

Salah satu perangkat lunak *Content Management System* yang dikenal luas yaitu MediaWiki, perangkat lunak yang dipakai di Wikipedia dan proyek-proyek sejenis.

## ***2.8 Metode Pengembangan Sistem***

### **2.1.2 Pengertian Pengembangan Sistem**

Pengembangan sistem (*systems development*) dapat berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Sistem yang lama perlu diperbaiki atau diganti disebabkan karena beberapa hal, yaitu sebagai berikut ini:

- Adanya permasalahan-permasalahan (problems) yang timbul di sistem yang lama.
- Ketidakterbacaan dalam sistem yang lama menyebabkan sistem yang lama tidak dapat beroperasi sesuai dengan yang diharapkan.
- Kecurangan-kecurangan disengaja yang menyebabkan tidak amannya harta kekayaan perusahaan dan kebenaran dari data menjadi kurang terjamin.
- Kesalahan-kesalahan yang tidak disengaja yang juga dapat menyebabkan kebenaran dari data kurang terjamin.
- Tidak efisiennya operasi.
- Tidak ditaatinya kebijaksanaan manajemen yang telah ditetapkan.
- Pertumbuhan organisasi.
- Sebuah sistem informasi adalah untuk mengatur manusia dan komponen-komponen mesin, dan prosedur-prosedur yang saling berkaitan untuk mendukung kebutuhan informasi atau bisnis pada sebuah organisasi dan para pengguna sistem. Sistem tersebut tidak seperti paket program perangkat lunak aplikasi tetapi harus terlebih dahulu dikostumisasi.

2. *Analysis* Tahap ini merupakan tahap dalam mencari informasi sebanyak-banyaknya mengenai sistem yang diteliti dengan melakukan metode-metode pengumpulan data sehingga ditemukan kelebihan dan kekurangan sistem serta *user requirement*. Tahap ini juga dilakukan untuk mencari pemecah masalah dan menganalisa bagaimana sistem akan dibangun untuk memecahkan masalah pada sistem sebelumnya.
3. *Design* Tahap ini merupakan tahapan perancangan sistem yang didalamnya dilakukan pemodelan sistem dengan *use case*, *activity diagram*, dan *sequence diagram*.
4. *Coding* Tahap ini merupakan tahapan dalam pengimplementasian sistem yang sudah dirancang dan dilakukan pengujian secara satuan, agar dapat mengetahui kesalahan-kesalahan yang terdapat dalam sistem dan segera dilakukan perbaikan.
5. *Testing* Tahap ini merupakan tahap pengujian sistem secara keseluruhan. Tahap ini sistem yang akan dikembangkan menggunakan teknik pengujian *black box*.
6. *Maintenance* Tahap ini merupakan tahapan penggunaan sistem oleh *user* yang didalamnya harus ada pemeliharaan sistem untuk menjaga proses 20 operasional sistem dan memungkinkan untuk dilakukan pengembangan sistem di kemudian hari.

#### **A. Keterlibatan Pengguna dalam Pengembangan Sistem**

Berikut ini beberapa contoh bagaimana keterlibatan pengguna di dalam pengembangan suatu sistem, yaitu:

- a. Pentingnya bagi pengguna untuk menjelaskan bagaimana sistem yang sedang berjalan pada bagian tempat pengguna bekerja.
- b. Menemukan dan mendiskusikan permasalahan yang dihadapi dan bagaimana hal itu dapat diperbaiki pada sistem yang baru.
- c. Kemungkinan perlu untuk memakai tenaga analis sistem dan desainer yang dibutuhkan dalam pengembangan sistem untuk memenuhi kebutuhan bagian.
- d. Kemungkinan anda sebagai pengguna selalu dilibatkan di dalam hal persetujuan proyek dan anggaran sebagai anggota *special steering committee*.
- e. Pada saat pengembangan sistem akan selesai, pengguna akan dimintai bantuannya untuk mengevaluasi dan uji coba, untuk memastikan bahwa sistem bekerja dengan sempurna.
- f. Anda sebagai pengguna turut membantu mempersiapkan sebagian dari dokumentasi yang dikumpulkan selama proses pengembangan sistem.
- g. Anda seharusnya menghadiri pengarahannya singkat dan sesi pelatihan untuk belajar bagaimana sistem baru akan mempengaruhi pekerjaan anda dan operasi prosedur baru nantinya.
- h. Terakhir tetapi pasti, anda akan menggunakan sistem yang baru tersebut.

## **2.8.1 Metode Pengembangan Sistem Informasi**

### **A. Metode System Development Life Cycle (SDLC)**

Model SDLC atau Sekuensial Linier sering disebut juga Model Air Terjun. Model ini mengusulkan sebuah pendekatan perkembangan perangkat

lunak yang sistematis dan sekuensial yang dimulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, pengujian, dan pemeliharaan

Model ini disusun bertingkat, setiap tahap dalam model ini dilakukan berurutan, satu sebelum yang lainnya. Model ini biasanya digunakan untuk membuat sebuah software dalam skala besar dan yang akan dipakai dalam waktu yang lama. Sangat cocok untuk pengembangan sistem yang besar. Tidak sesuai atau tidak terlalu disarankan untuk *small scale project* karena:

7. Resource intensive
8. Tidak fleksibel
9. Sulit untuk aplikasi dengan perubahan cara pengambilan keputusan yang cepat

#### **a. Tahapan-tahapan (SDLC):**

##### **1. Fase Perencanaan Sistem**

Dalam tahapan ini dibentuk suatu struktur kerja strategis yang luas dan pandangan sistem informasi baru yang jelas yang akan memenuhi kebutuhan-kebutuhan pemakai informasi. Proyek sistem dievaluasi dan dipisahkan berdasarkan prioritasnya. Proyek dengan prioritas tertinggi akan dipilih untuk pengembangan. Penyediaan sumber daya baru dan penyediaan dana untuk pengembangan sistem. Rencana kerja yang matang juga disusun untuk menjalankan tahapan-tahapan lainnya. Hasil dari tahapan ini adalah : Langkah-langkah detail rencana kerja dan penugasan untuk anggota tim.

##### **2. Fase Analisis Sistem**

- Dilakukan proses penilaian, identifikasi dan evaluasi komponen dan hubungan timbal-balik yang terkait dalam pengembangan system: definisi masalah, tujuan, kebutuhan, prioritas dan kendala-

kendala system, ditambah identifikasi biaya, keuntungan dan estimasi jadwal untuk solusi yang berpotensi.

- Tim proyek sistem memperoleh pengertian yang lebih jelas tentang alasan untuk mengembangkan suatu sistem baru.
- Laporan yang dihasilkan menyediakan suatu landasan untuk membentuk suatu tim dan memulai fase analisis sistem.
- Ruang lingkup analisis sistem ditentukan pada fase ini. Profesional sistem mewawancarai calon pemakai dan bekerja dengan pemakai yang bersangkutan untuk mencari penyelesaian masalah dan menentukan kebutuhan pemakai.
- Fase analisis sistem adalah fase profesional sistem melakukan kegiatan analisis sistem.
- Beberapa aspek sistem yang sedang dikembangkan mungkin tidak diketahui secara penuh pada fase ini, jadi asumsi kritis dibuat untuk memungkinkan berlanjutnya siklus hidup pengembangan sistem.
- Pada akhir fase analisis sistem, laporan analisis sistem disiapkan. Laporan ini berisi penemuan-penemuan dan rekomendasi. Bila laporan ini disetujui, tim proyek sistem siap untuk memulai fase perancangan sistem secara umum. Bila laporan tidak disetujui, tim proyek sistem harus menjalankan analisis tambahan sampai semua peserta setuju

### **3. Fase Perancangan Sistem secara Umum**

10. Dibentuk alternatif-alternatif perancangan konseptual untuk pandangan pemakai. Alternatif ini merupakan perluasan kebutuhan pemakai. Alternatif perancangan konseptual memungkinkan manajer dan pemakai untuk memilih rancangan terbaik yang cocok untuk kebutuhan mereka.



11. Pada fase ini analis sistem mulai merancang proses dengan mengidentifikasi laporan-laporan dan output yang akan dihasilkan oleh sistem yang diusulkan. Data masing-masing laporan ditentukan. Biasanya, perancang sistem membuat sketsa form atau tampilan yang mereka harapkan bila sistem telah selesai dibentuk. Sketsa ini dilakukan pada kertas atau pada tampilan komputer.

#### **4. Fase Evaluasi dan Seleksi Sistem**

Akhir fase perancangan sistem secara umum menyediakan point utama untuk keputusan investasi. Oleh sebab itu dalam fase evaluasi dan seleksi sistem ini nilai kualitas sistem dan biaya/keuntungan dari laporan dengan proyek system dinilai secara hati-hati dan diuraikan dalam laporan evaluasi dan seleksi sistem.

Jika tak satupun alternatif perancangan konseptual yang dihasilkan pada fase perancangan sistem secara umum terbukti dapat dibenarkan, maka semua alternatif akan dibuang. Biasanya, beberapa alternatif harus terbukti dapat dibenarkan, dan salah satunya dengan nilai tertinggi dipilih untuk pekerjaan akhir. Bila satu alternatif perancangan sudah dipilih, maka akan dibuatkan rekomendasi untuk sistem ini dan dibuatkan jadwal untuk perancangan detailnya.

#### **5. Fase Perancangan Sistem secara Detail**

Pada fase ini semua komponen dirancang dan dijelaskan secara detail. Perencanaan output (layout) dirancang untuk semua layar, form-form tertentu dan laporan-laporan yang dicetak. Semua output direview dan disetujui oleh pemakai dan didokumentasikan.

Berdasarkan perancangan output dan input, proses-proses dirancang untuk mengubah input menjadi output. Transaksi-transaksi dicatat dan dimasukkan secara online atau batch. Macam-macam

model dikembangkan untuk mengubah data menjadi informasi. Prosedur ditulis untuk membimbing pemakai dan pesonel operasi agar dapat bekerja dengan sistem yang sedang dikembangkan.

Database dirancang untuk menyimpan dan mengakses data. Kendali-kendali yang dibutuhkan untuk melindungi sistem baru dari macam-macam ancaman dan error ditentukan.

Pada akhir fase ini, laporan rancangan sistem secara detail dihasilkan. Laporan ini mungkin berisi beribu-ribu dokumen dengan semua spesifikasi untuk masing-masing rancangan sistem yang terintegrasi menjadi satu kesatuan. Laporan ini dapat juga dijadikan sebagai buku pedoman yang lengkap untuk merancang, membuat kode dan menguji sistem; instalasi peralatan; pelatihan; dan tugas-tugas implementasi lainnya.

## **6. Fase Implementasi Sistem dan Pemeliharaan Sistem**

12. Sistem siap untuk dibuat dan diinstalasi.

13. Sejumlah tugas harus dikoordinasi dan dilaksanakan untuk implementasi sistem baru.

14. laporan implementasi yang dibuat pada fase ini ada dua bagian, yaitu:

- Rencana implementasi dalam bentuk *Gantt Chart* atau *Program and Evaluation Review Technique (PERT) Chart*
- Penjadwalan proyek dan teknik manajemen. Bagian kedua adalah laporan yang menerangkan tugas penting untuk melaksanakan implementasi sistem, seperti :

15. Pengembangan perangkat lunak

16. Persiapan lokasi peletakkan system

17. Instalasi peralatan yang digunakan

## 18. Pengujian Sistem

### b. Kekurangan dan Kelebihan

#### 1. Kelebihan

- Mudah diaplikasikan.
- Memberikan template tentang metode analisis, desain, pengkodean, pengujian, dan pemeliharaan.

#### 2. Kekurangan

- Jarang sekali proyek riil mengikuti aliran sekuensial yang dianjurkan model karena model ini bisa melakukan iterasi tidak langsung.
- Pelanggan sulit untuk menyatakan kebutuhan secara eksplisit sehingga sulit untuk mengakomodasi ketidakpastian pada saat awal proyek.
- Pelanggan harus bersikap sabar karena harus menunggu sampai akhir proyek dilalui. Sebuah kesalahan jika tidak diketahui dari awal akan menjadi masalah besar karena harus mengulang dari awal.
- Pengembang sering melakukan penundaan yang tidak perlu karena anggota tim proyek harus menunggu tim lain untuk melengkapi tugas karena memiliki ketergantungan hal ini menyebabkan penggunaan waktu tidak efisien.

## 2.9 Apache HTTP Server

*Server HTTP Apache atau Server Web/WWW Apache* merupakan server web yang dapat dijalankan di banyak sistem operasi Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta *platform* lainnya, yang berguna untuk melayani dan memfungsikan situs web.

Protokol yang digunakan untuk melayani fasilitas *web/www* ini menggunakan *HTTP*.

*Apache* memiliki fitur-fitur canggih seperti pesan kesalahan yang dapat dikonfigur, autentikasi berbasis basis data dan lain-lain. *Apache* juga didukung oleh sejumlah antarmuka pengguna berbasis grafik (GUI) yang memungkinkan penanganan server menjadi mudah.

*Apache* adalah perangkat lunak sumber terbuka dikembangkan oleh komunitas terbuka yang terdiri dari pengembang-pengembang di bawah naungan *Apache Software Foundation*.

*Apache* adalah komponen server web dari paket perangkat lunak *LAMP* (*Linux*, *Apache*, *MySQL*, *PHP/Perl/bahasa pemrograman Python*).

## **2.10 UML**

*Unified Modeling Language (UML)* merupakan bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak.

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

- *Use Case Diagram* untuk memodelkan proses bisnis.
- *Conceptual Diagram* untuk memodelkan konsep-konsep yang ada di dalam aplikasi.
- *Sequence Diagram* untuk memodelkan pengiriman pesan (*message*) antar *objects*.
- *Collaboration Diagram* untuk memodelkan interaksi antar *objects*.
- *State Diagram* untuk memodelkan perilaku *objects* di dalam sistem.

- Activity Diagram untuk memodelkan perilaku *Use Cases* dan *objects* di dalam *system*.
- Class Diagram untuk memodelkan struktur kelas.
- Object Diagram untuk memodelkan struktur *object*.
- Component Diagram untuk memodelkan komponen *object*.
- Deployment Diagram untuk memodelkan distribusi aplikasi.

## **2.11 Web Browser**

adalah perangkat lunak yang berfungsi untuk menerima dan menyajikan sumber informasi dari Internet. Sebuah sumber informasi diidentifikasi dengan pengidentifikasi sumber seragam yang dapat berupa halaman web, gambar, video, atau jenis konten lainnya

Meskipun penjelajah web terutama ditujukan untuk mengakses Internet, sebuah penjelajah juga dapat digunakan untuk mengakses informasi yang disediakan oleh server web dalam jaringan pribadi atau berkas pada sistem berkas. Beberapa penjelajah web yang populer adalah Google Chrome, Firefox, Internet Explorer, Opera, dan Safari.

Penjelajah web bisa dibedakan lewat fitur-fitur yang disediakan. Penjelajah modern dan halaman web biasanya menggunakan banyak fitur dan teknik yang tidak ada pada masa-masa awal web. Disebabkan adanya perang penjelajah web, fitur-fitur web dan penjelajah web semakin cepat dikembangkan.

Berikut daftar beberapa elemen dan fitur-fitur tersebut:

- ActiveX.
- Autocompletion (Pengisian otomatis) URL dan formulir data.

- Markah buku untuk mengikuti lokasi yang sering diakses.

- Cascading Style Sheets (CSS).
- Kuki yang membolehkan sebuah website untuk mengetahui seorang pengguna lama.
- Tembolok web - Halaman web "disimpan" dalam memori ketika kita membukanya agar dapat diakses lagi walaupun sedang luring.
- Digital certificate (Sertifikat Digital).
- Pemuatan gambar menggunakan format gambar yang sudah terkenal seperti GIF, PNG, JPEG, SVG.
- Flash.
- Favicon.
- Font, ukuran, warna.
- Formulir untuk mengirimkan informasi.
- Frame dan IFrames.
- Gambar.
- Integrasi dengan aplikasi *desktop* lainnya.
- Offline browsing (Penjelajah Tertutup) terhadap isi web yang sudah disimpan terlebih dahulu.
- Java applet.
- JavaScript untuk isi yang lebih dinamis.
- Pengaturan pengunduhan.
- Penyaringan iklan (*Ad filtering*).
- Plug-in.
- Sejarah kunjungan ke halaman-halaman web terakhir.
- Session management.

- Tabbed browsing.
- Tabel.
- XHTML dan XML.



- DHTML.
- HTTPS.

## ***2.12 Sistem***

adalah suatu kesatuan yang terdiri atas komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi, atau energi untuk mencapai suatu tujuan. Istilah ini sering digunakan untuk menggambarkan suatu set entitas yang berinteraksi, di mana suatu model matematika seringkali bisa dibuat.

Sistem merupakan elemen yang memiliki unsur seperti masukan (*input*), pengolahan (*processing*), dan keluaran (*output*). Dari unsur-unsur tersebut saling berkaitan untuk mencapai suatu tujuan tertentu

## **Bab 3**

---

### **ANALISIS DAN PERANCANGAN**

---

#### **3.1 Analisis**

Analisis merupakan penelaahan atau penelitian dengan melakukan suatu percobaan yang menghasilkan kesimpulan dari sebuah sistem informasi. Analisis spesifikasi sistem kebutuhan ini yang berkaitan dengan sistem aplikasi ini yaitu, analisa sistem yang berjalan, analisis deskripsi dokumentasi yang digunakan, kebutuhan penggunaan sistem, dan kebutuhan perangkat lunak dan perangkat keras yang digunakan, dengan tujuan untuk mengidentifikasi segala permasalahan atau hambatan-hambatan yang terjadi. Sehingga dapat diadakannya perbaikan-perbaikan yang dapat meningkatkan kualitas dari sistem informasi yang akan dibuat.

Analisis merupakan tahap yang kritis yang sangat penting karena suatu kesalahan dalam tahap ini akan berpengaruh pada tahap berikutnya. Penelitian ini membuktikan bahwa kesalahan yang diperbaiki setelah tahap analisis akan memakan biaya yang lebih besar dari pada jika diperbaiki saat dilakukan analisis.

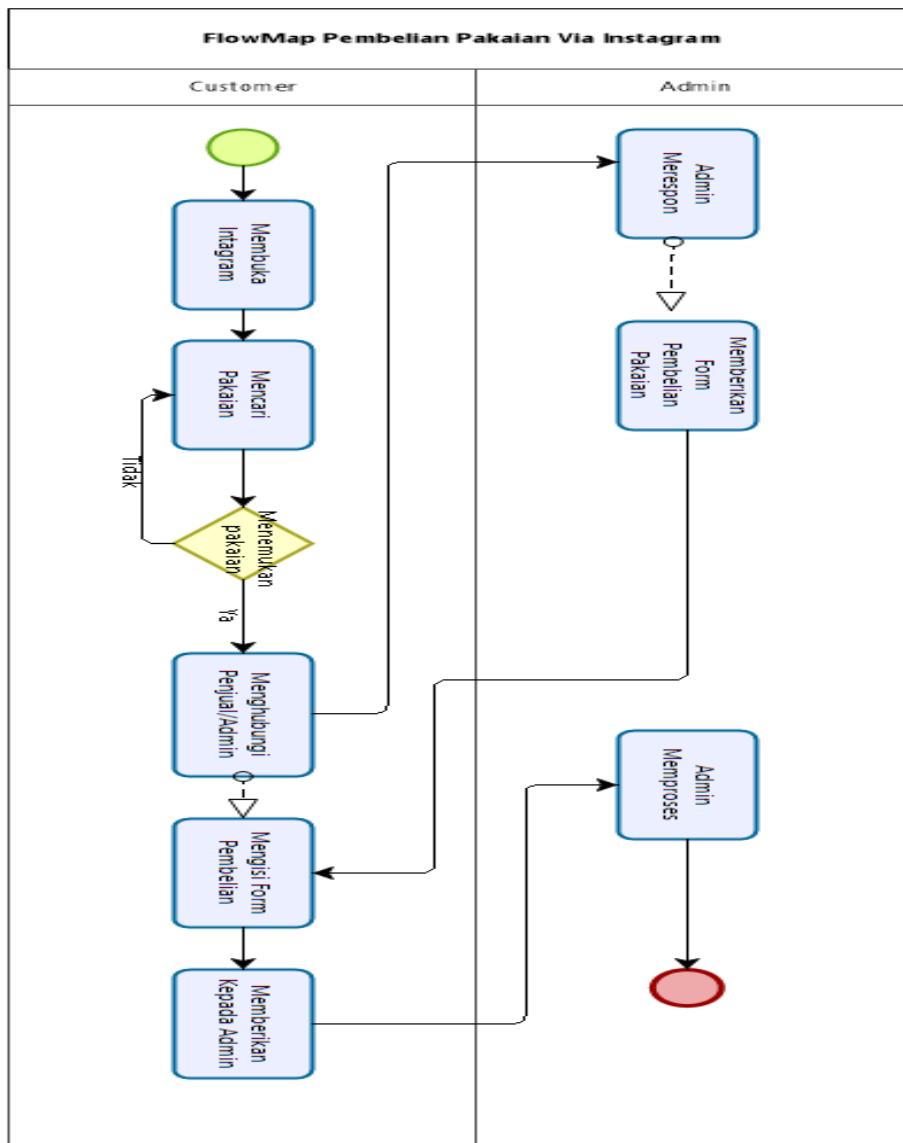
Langkah-langkah yang harus dilakukan dalam tahap analisis sistem, yaitu:

1. Pengenalan atau identifikasi masalah, langkah ini merupakan langkah awal yang harus dilakukan, yaitu dengan mengidentifikasi permasalahan yang ada sehingga sasaran yang ingin dicapai dapat terlaksana.
2. Memahami kerja dari sistem yang ada, langkah ini dilakukan dengan mempelajari secara rinci bagaimana jalannya sistem yang sudah ada.
3. Menganalisis hasil penelitian, hal yang perlu diperhatikan dalam tahapan ini adalah menganalisis kebutuhan informasi pemakai sistem berdasarkan data yang diperoleh atas dasar hasil penelitian.
4. Membuat laporan penelitian, merupakan tahap akhir yang disusun dalam suatu rangkuman dari langkah-langkah sebelumnya.<sup>[6]</sup>

### 3.2 Analisis Sistem yang sedang Berjalan

#### 3.2.1 FlowMapp Pembelian Pakaian Via Instagram

Berikut ini adalah analisis sistem yang sedang berjalan dalam bentuk *flow map* atau prosedur pembelian pakaian pada saat ini:

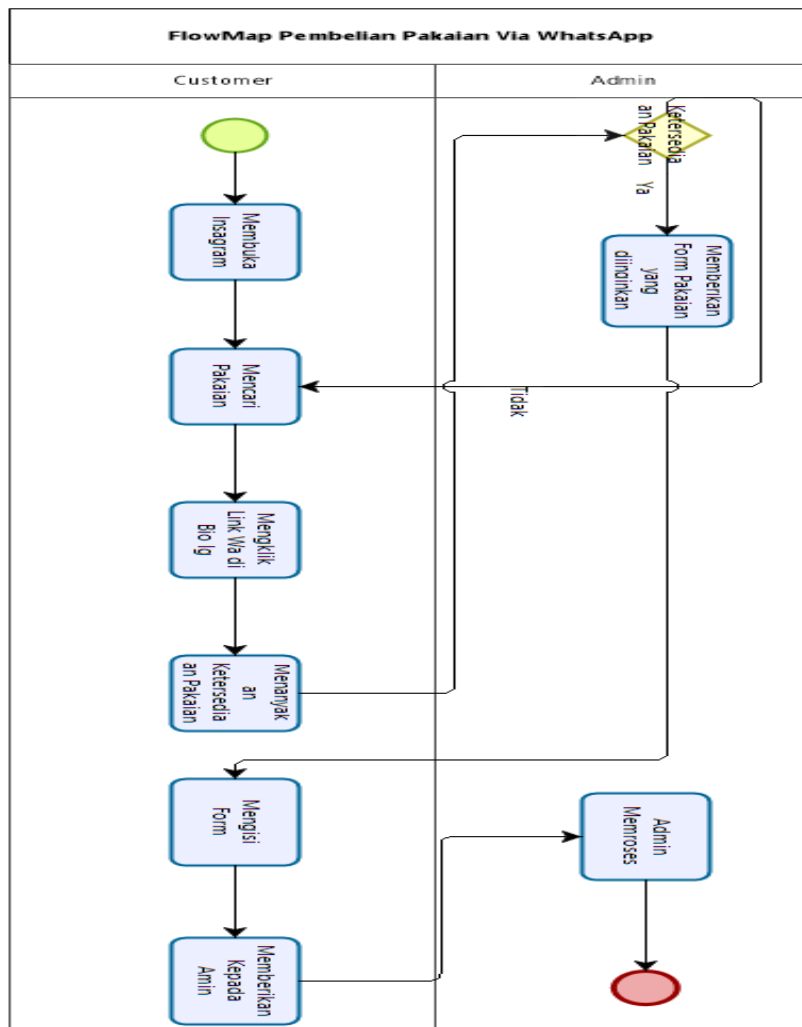


Gambar 3.1 *FlowMap* Pembelian Via Instagram

**Penjelasan :**

1. Customer membuka Instagram.
2. Customer mengklik Web yang ada pada bio instagram
3. Customer mencari pakaian yang akan dibeli
4. Customer memasukkan ke dalam keranjang pembelian
5. Melakukan pembelian, dan mengisi data
6. Jika tidak jadi membeli maka akan Msuk ke pencarian pakaian
7. Setelah mendapatkan pakaian, customer menghubungi admin Instagram
8. Admin memproses pesanan
9. Transaksi selesai.

### 3.2.2 Flowmap Pembelian Via WhatsApp



Gambar 3.2 *FlowMap* Pembelian Via WhatsApp

**Penjelasan :**

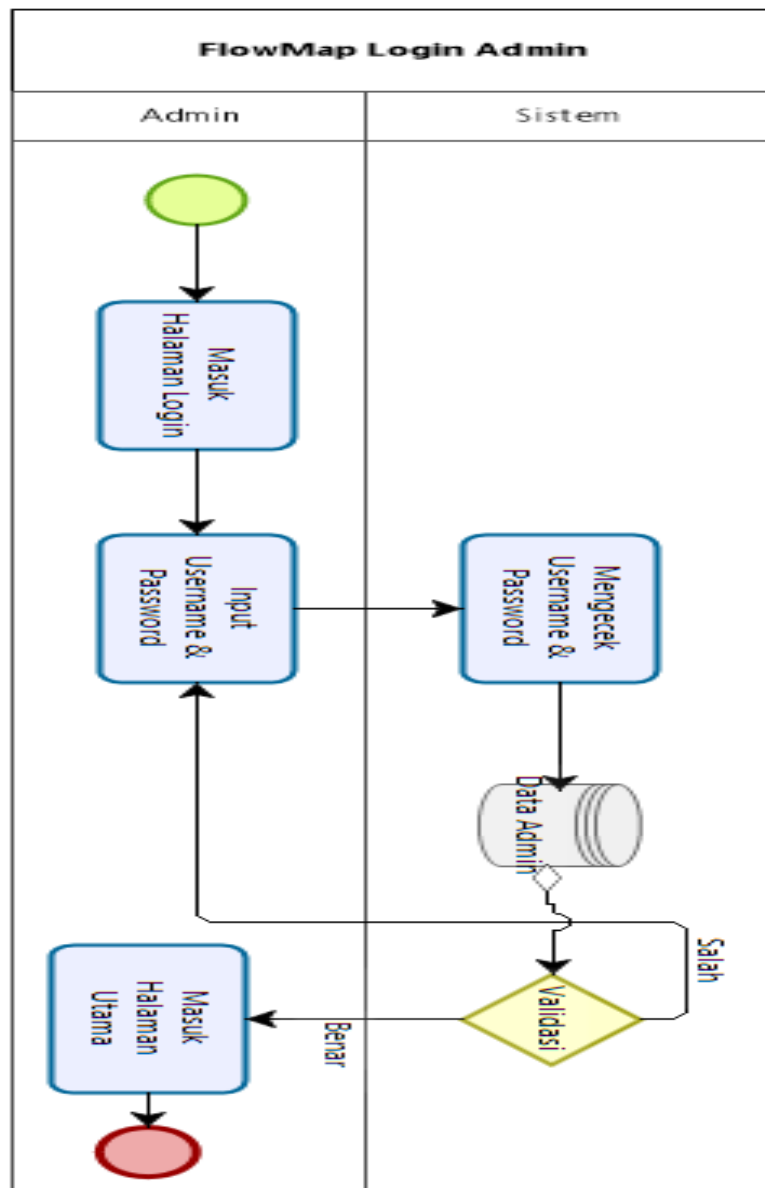
1. Customer Membuka Instagram
2. Customer Mencari Pakaian
3. Customer Mengklik Link WA di Bio Instagram
4. Customer menanyakan ketersediaan pakaian
5. Admin Melihat ketersediaan pakaian

6. Jika Tidak ada maka admin memberikan dform pakaian yang diinginkan
7. Customer mengisi Form
8. Customer memberikan memberikan data kepada admin
9. Admin melakukan proses
10. Proses selesai

### **3.3 Analisis Sistem yang akan Dibangun**

Sistem layanan informasi ini akan mengubah proses pembelian pakaian yang sebelumnya pembelian hanya bisa melalui Instagram sehingga bisa melakukan pembelian melalui website yang terhubung dengan Instagram. Berikut ini terdapat beberapa prosedur *flowmap* dari sistem yang akan dibangun:

#### **3.3.1 FlowMap Login Admin**



Gambar 3.3 *FlowMap Login Admin*

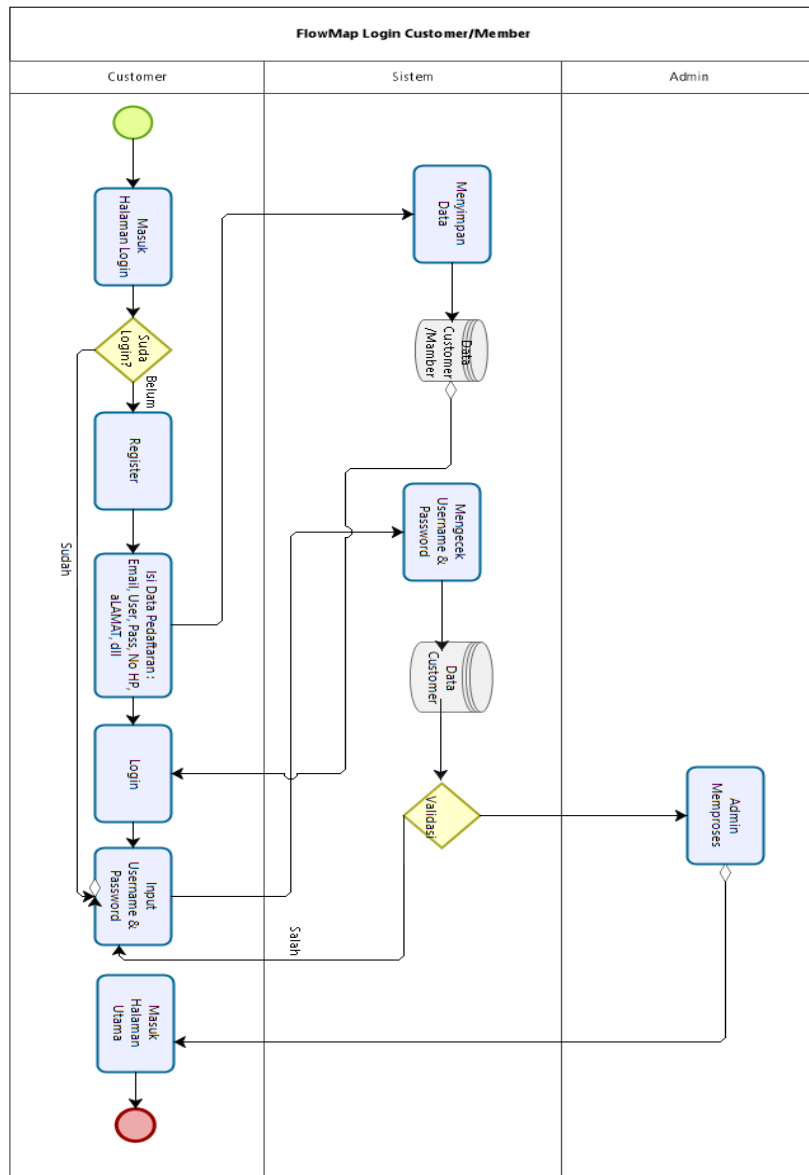
**Penjelasan :**

1. Admin masuk kehalaman login
2. Admin mengisi username & password
3. Sistem mengecek username & password
4. Data admin masuk ke database sistem



5. Sistem melakukan validasi
6. Jika salah maka harus mengisi username & password lagi
7. Jika benar lanjut ke halaman utama

### 3.3.2 FlowMap Login Customer dan Member

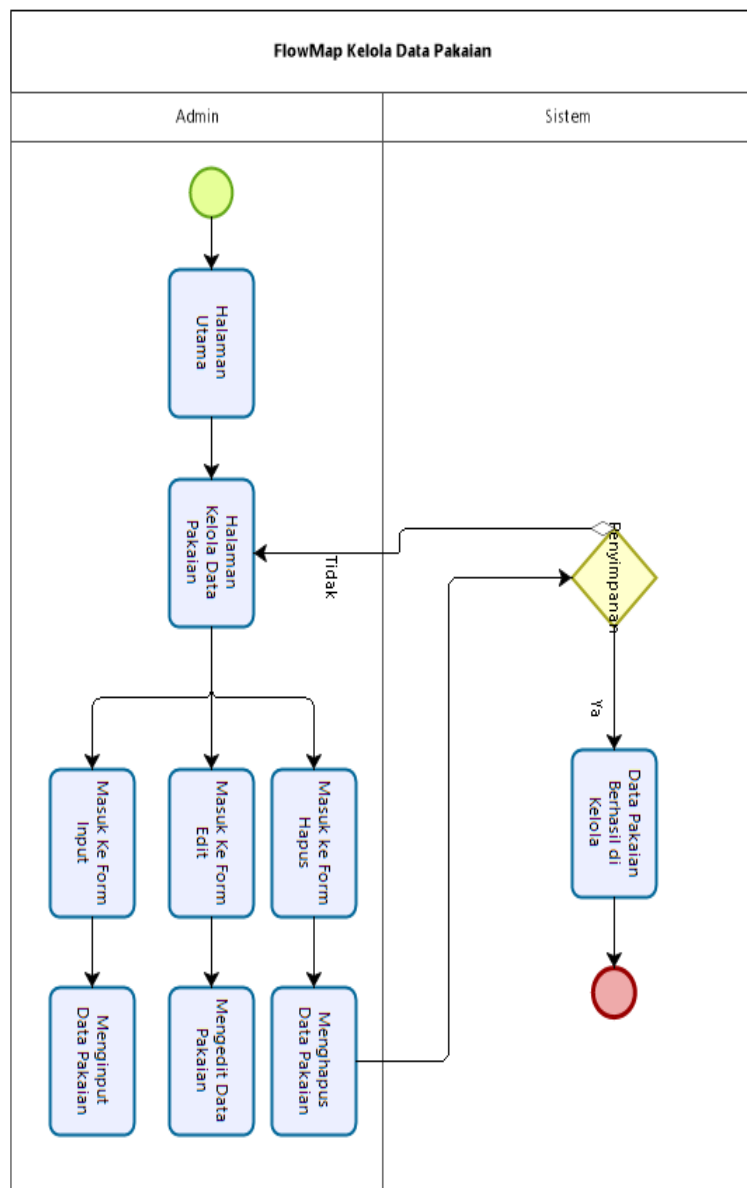


Gambar 3.4 *FlowMap Login Customer dan Member*

Penjelasan :

1. admin masuk halaman login
2. apakah sudah login?
3. Jika belum maka register
4. Lalu isi data, email, user, pass, nohp, dll
5. Sistem menyimpan data ke dalam database
6. Jika sudah langsung login
7. Admin masuk ke halaman login
8. Admin mengisi username & password
9. Sistem mengecek username & password
10. Data admin masuk ke database sistem
11. Sistem melakukan validasi
12. Jika salah maka harus mengisi username & password lagi
13. Jika benar lanjut ke halaman utama

### **3.3.3 FlowMap Kelola Data Pakaian**



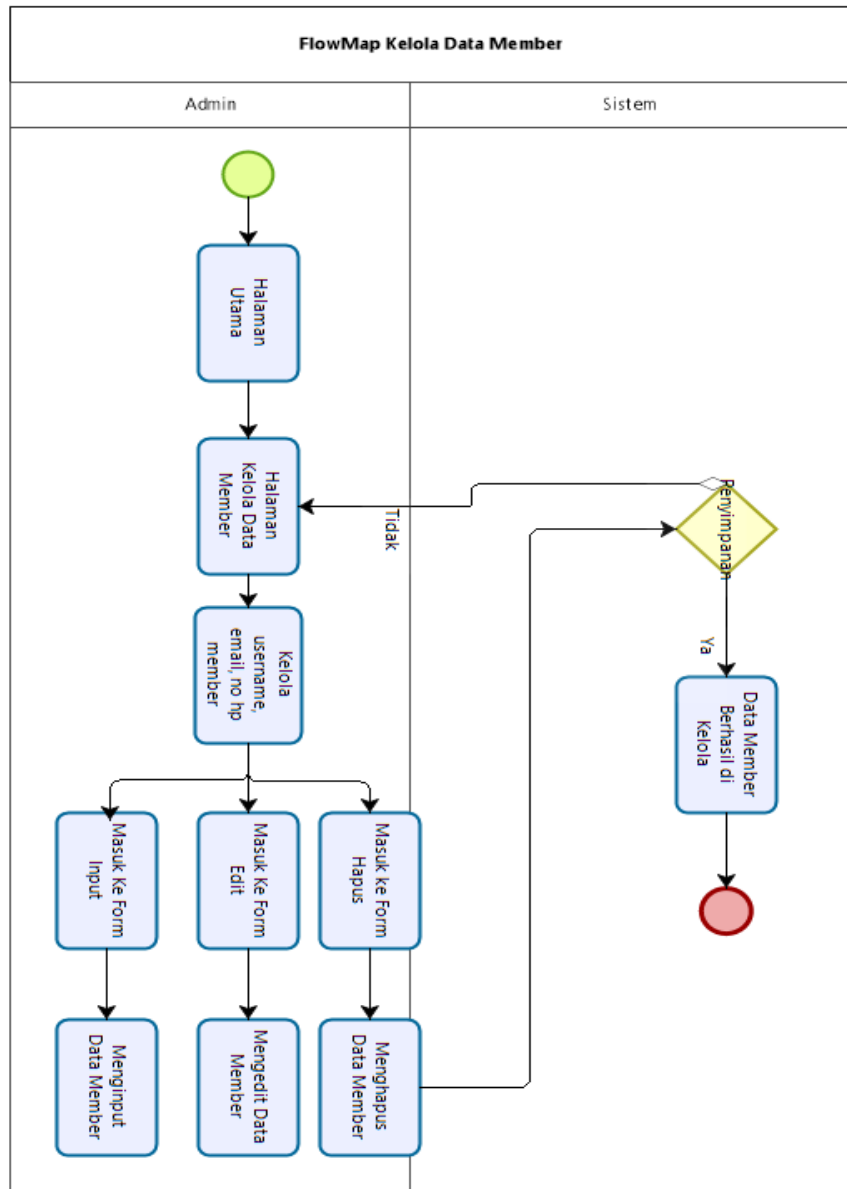
Gambar 3.5 *FlowMap* Kelola Data Pakaian

Penjelasan :

1. Admin masuk ke halaman utama

2. Admin masuk ke halaman kelola data pakaian
3. Admin masuk ke form Hapus, Edit, Input
4. Admin melakukan Hapus, Edit, Input pakaian
5. Sistem melakukan penyimpanan
6. Jika tidak maka akan kembali ke halaman data pakaian
7. Jika ya maka proses selesai

#### **3.3.4 Flowmap Kelola Data Member**



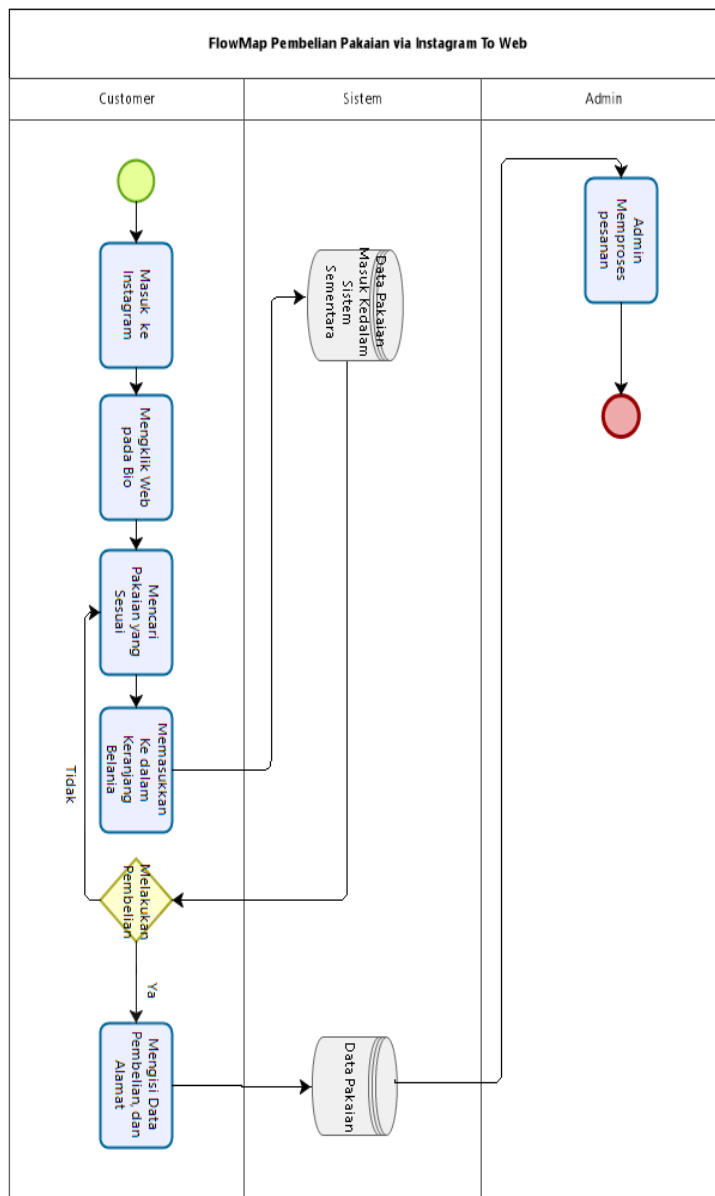
Gambar 3.6 *FlowMap Kelola Data Member*

### Penjelasan 5

1. Admin masuk ke halaman utama
2. Admin masuk ke halaman kelola data pakaian
3. Admin Kelola uername, email, nohp member
4. Admin masuk ke form Hapus, Edit, Input

5. Admin melakukan Hapus, Edit, Input pakaian
6. Sistem melakukan penyimpanan
7. Jika tidak maka akan kembali ke halaman data pakaian
8. Jika ya maka proses selesai

### 3.3.5 FlowMap Pembelian Pakaian Via Ig To Web

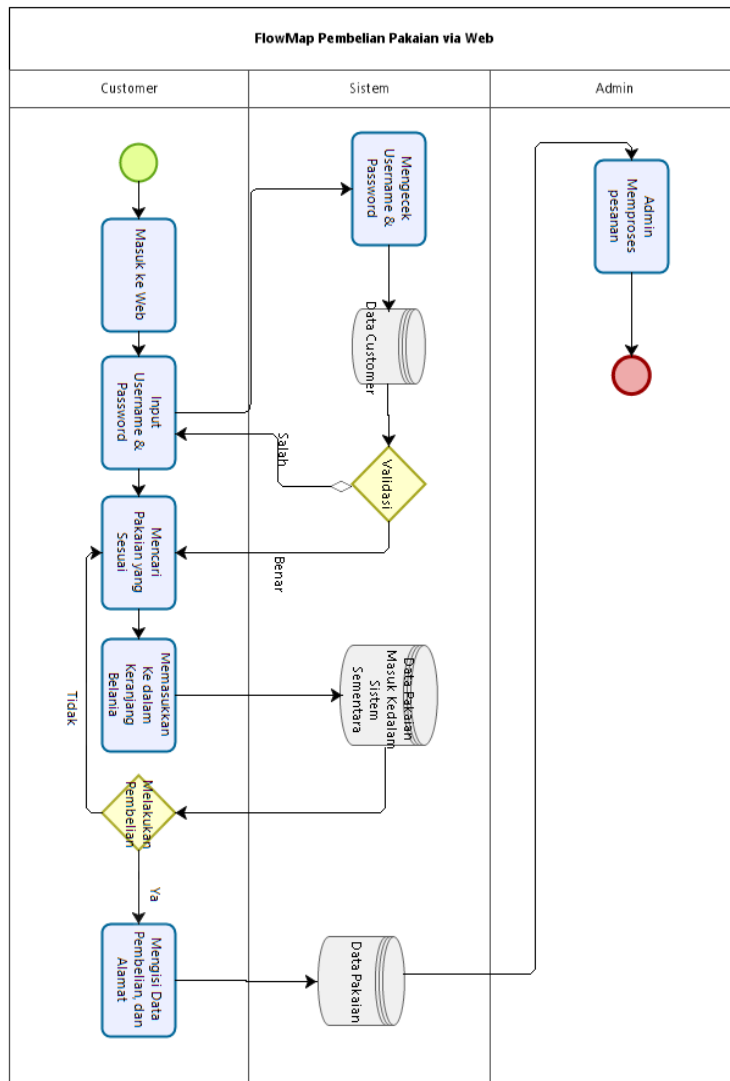


Gambar 3.7 *FlowMap* Pembelian Pakian Via Ig To Web

**Penjelasan :**

1. Customer masuk ke instagram
2. Customer mengklik web pada bio
3. Customer mencari pakaian yang diinginkan
4. Customer memasukan ke dalam keranjang belanja
5. Data yang masuk di keranjang akan masuk ke database sistem sementara, karena customer belum tentu jadi membeli pakaian
6. Customer melakukan pembelian
7. Jika tidak maka akan masuk kembali ke pencarian
8. Customer mengisi data pembelian dan alamat
9. Masuk ke database sistem
10. Admin memproses pesanan

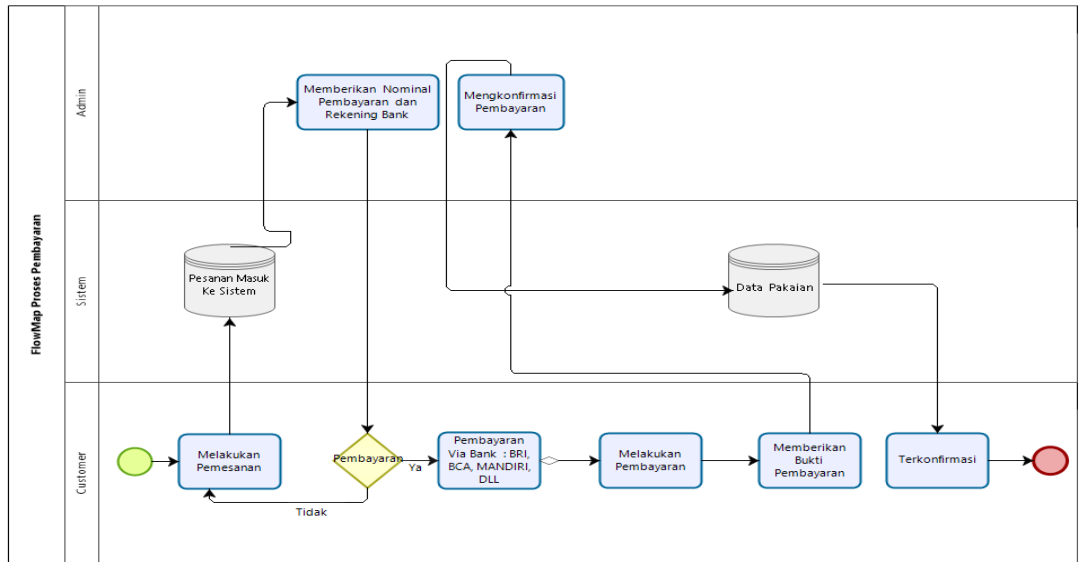
**3.3.6 Flowmap Pembelian Via Web**



Gambar 3.8 *Flowmap* Pembelian Via Web

### 3.3.7 Flowmap Pembayaran





Gambar 3.9 *FlowMap* Pembayaran Pakaian

#### Penjelasan :

1. Customer melakukan pemesanan
2. Pesanan masuk ke database sistem
3. Admin memberikan nominal pembayaran dan rekening Bank
4. Customer melakukan pembayaran Via Bank BRI, BCA, MANDIRI, dll
5. Jika tidak maka customer akan kembali ke pesanan
6. Jika ya maka melakukan pembayaran
7. Memberika bukti pembayaran kepada admin
8. Admin mengkonfirmasi pembayaran
9. Terkonfirmasi ke database dan customer
10. Proses selesai

#### 3.3.8 **FlowMap Pengiriman**



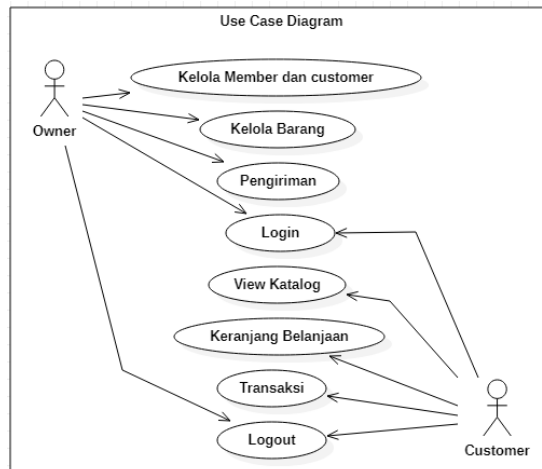
3. Admin melakukan pengiriman ke jasa kurir
4. Jasa kirim mengecek paket yang ingin dikirim
5. Jasa kirim memberikan nominal berapa bnhak biaya kirim
6. Admin membayar biaya kirim
7. Jasa kirim menerima pembayaran
8. Jasa kirim memberikan nomor pesanan
9. Admin menerima nomor pengiriman
10. Proses Pengiriman
11. Admin mengkonfirmasi pengiriman ke web belanja
12. Sistem memproses pengiriman
13. Data pengiriman masuk ke sistem
14. Customer menerima notifikasi barang sudah dikirim dan berapa lama waktu pengiriman
15. Customer dapat melacak status pengiriman
16. Pesanan sudah tiba dalam waktu yang ditentukan ?
17. Jika belum maka kurir masih dalam proses pengiriman
18. Jika sudah maka konfirmasi pesanan telah tiba
19. Memberikan nilai
20. Jasa kurir admin dan sistem terkonfirmasi brng telah tiba
21. Proses selesai

### **3.4 Perancangan**

Untuk mempermudah proses pengelolaan data barang maka dirancanglah sebuah sistem informasi *online* untuk membuat proses pengelolaan barang menjadi lebih efektif, memberikan kemudahan bagi petugas untuk melakukan pendataan, serta penempatan tataletak barang.

#### **3.4.1 Use Case Diagram**

Diagram *use case* menggambarkan interaksi antar *use case* dan aktor dalam satu sistem.



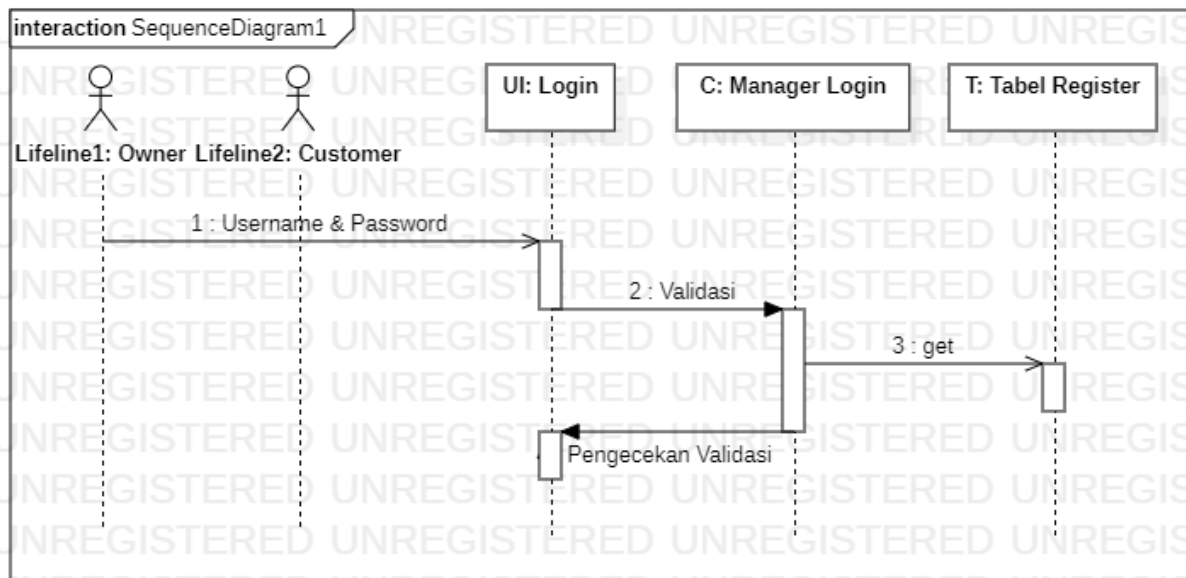
Gambar 3.11 *Use Case Diagram*

### 3.4.2 *Sequence Diagram*

*Sequence Diagram* merupakan penggambaran keterhubungan atau interaksi antar objek dalam suatu jangka waktu. *Sequence Diagram* terutama menampilkan interaksi Antara pengguna (*user*) dengan sistem.

#### 3.4.2.1 *Sequence Diagram Login*

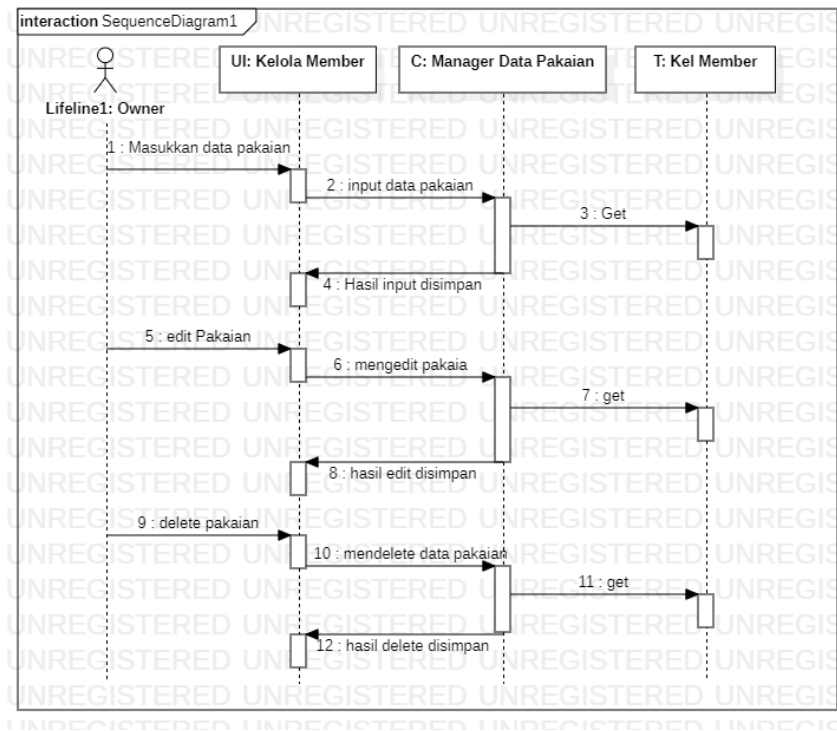
*Owner* dan *customer* menginput *username & password* untuk *login*, Sistem akan mengecek benar tidaknya *username & password* tersebut. Jika benar kemudia sistem akan masuk kehalaman utamam namun jka salah akan kembali kehalaman *login*.



Gambar 3.12 *Sequence Diagram Login*

#### 3.4.2.2 *Sequence Kelola Pakaian*

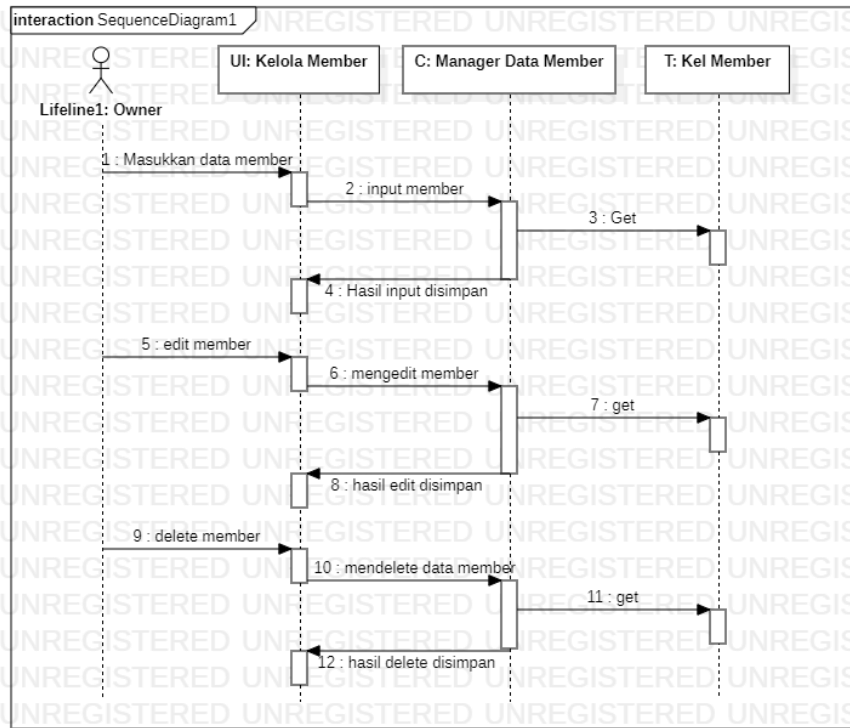
*Owner* masuk kedalam kelola pakaian, untuk melakukan *input* pakaian *owner* memilih menu *input* pakaian, kemudian memasukkan data pakaian, setelah selesai menginputkan data pakaian, data akan disimpan.



Gambar 3.13 *Sequence* kelola pakaian

### 3.4.2.3 *Sequence* Kelola Member

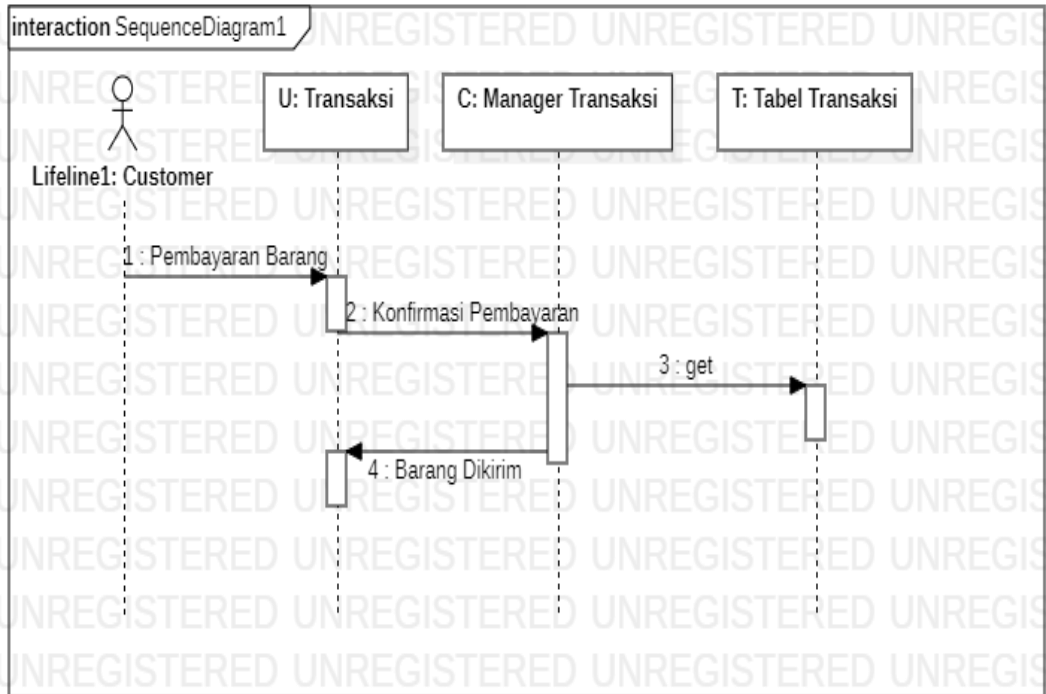
*Owner* masuk kehalaman kelola *member*, untuk melakukan *input member* petugas memilih menu *input member*, kemudian masukkan data *member*.



Gambar 3.14 *sequence kelola member.*

#### 3.4.2.4 Sequence Diagram Transaksi

Untuk pembelian barang, masuk kemenu utama, kemudian mencari barang yang akan dibeli, ketika ingin membeli barang, *customer* harus menginputkan *username* & *password*. Jika *username* & *password* benar langsung saja ke transaksi, dan jika tsalah akan kembali kehalaman cari barang.

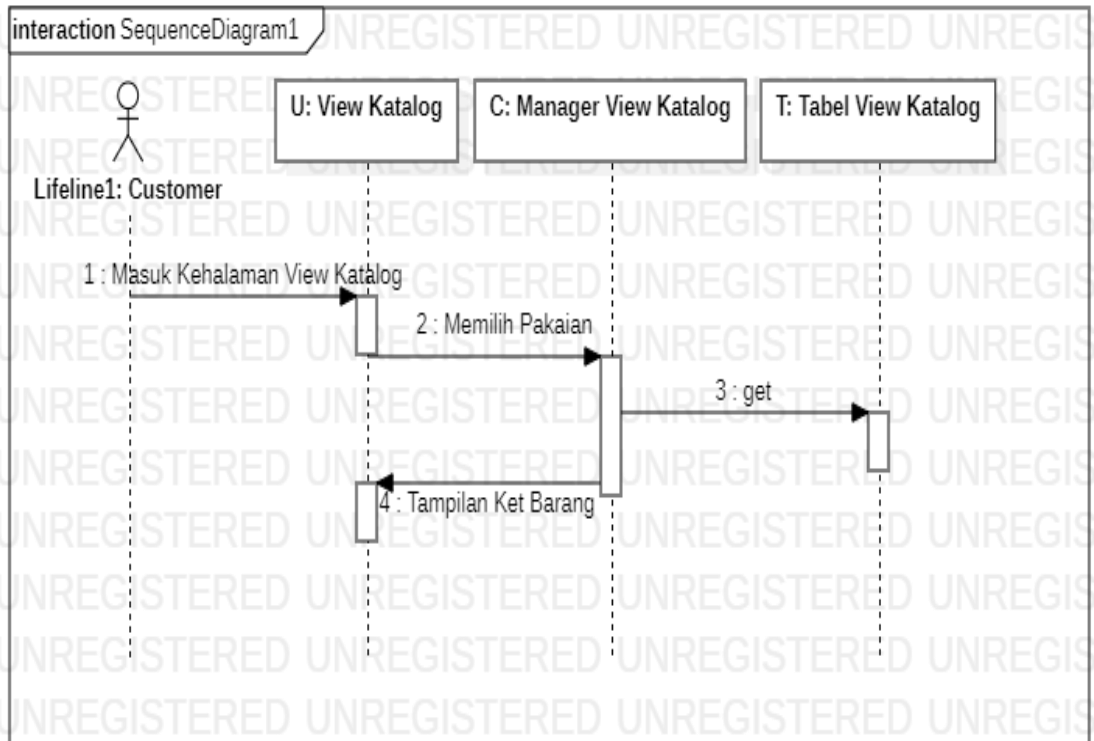


Gambar 3.15 *sequence transaksi*

#### 3.4.2.5 Sequence View Katalog

Untuk *view katalog*, *customer* memasuki halaman utama lalu memasuki *username* dan *password*, memilih barang, tampilan keterangan pakaian yang dibeli, melakukan pembayaran dan selesai.

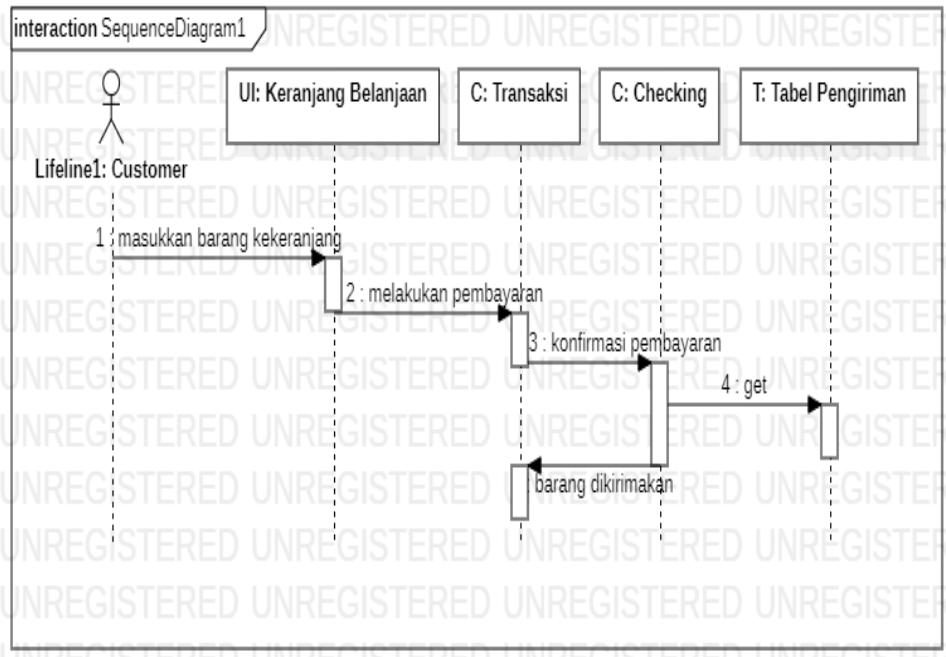




Gambar 3.16 *Sequence diagram view catalog*

#### 3.4.2.6 Sequence Keranjang Belanjaan

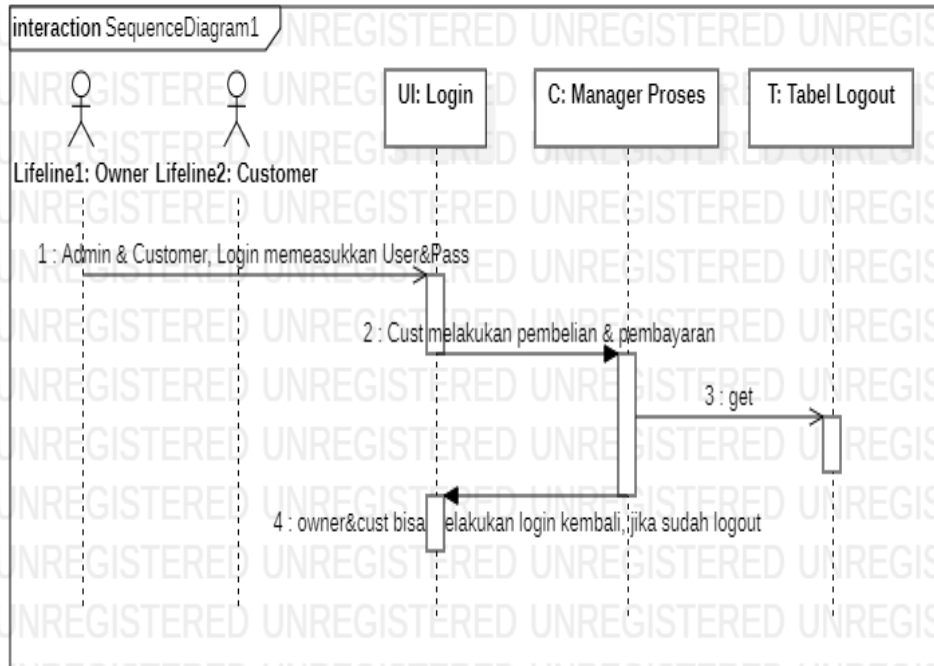
Setelah *customer* selesai memilih pakaian, lalu membeli pakaian yang dipilih dan melakukan pembayaran, jika selesai maka akan kembali kemenu pembelian.



Gambar 3.17 *sequence diagram* pembelian

#### 3.4.2.7 Sequence Logout

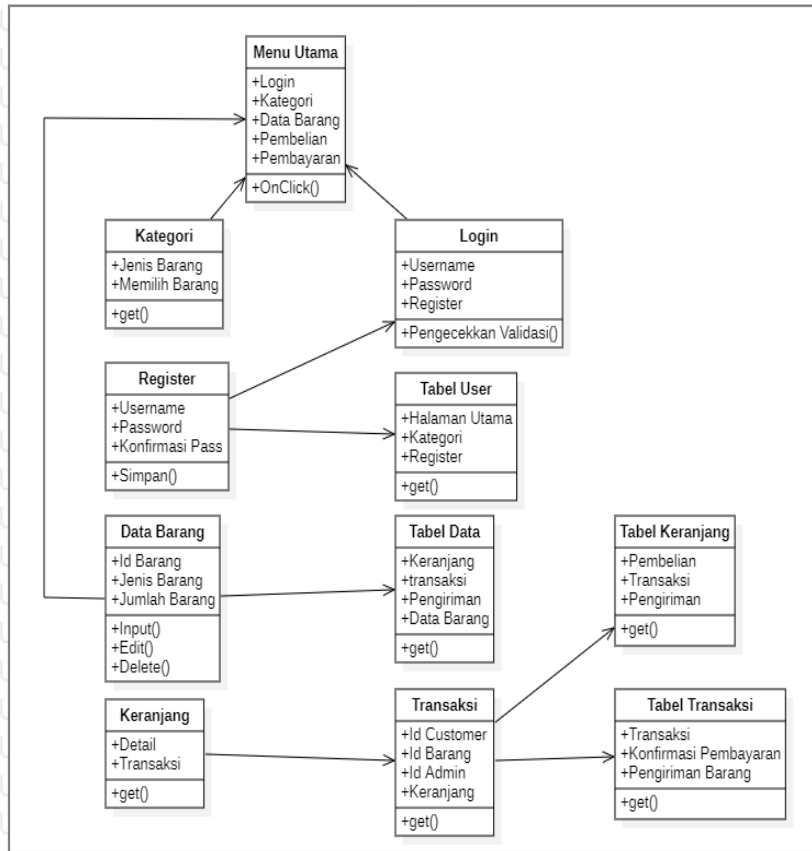
Setelah melakukan semua yang diinginkan maka bisa *logout* aplikasi, dan akan kembali ke halaman *login*.



Gambar 3.18 *sequence diagram logout*

### 3.4.3 *Class Diagram*

*Class diagram* merupakan struktur tetap yang akan digunakan dalam sebuah aplikasi. Dalam proses pembuatan aplikasi ini, *class diagram* yang akan digunakan sebagai berikut:



Gambar 3.19 Class Diagram.

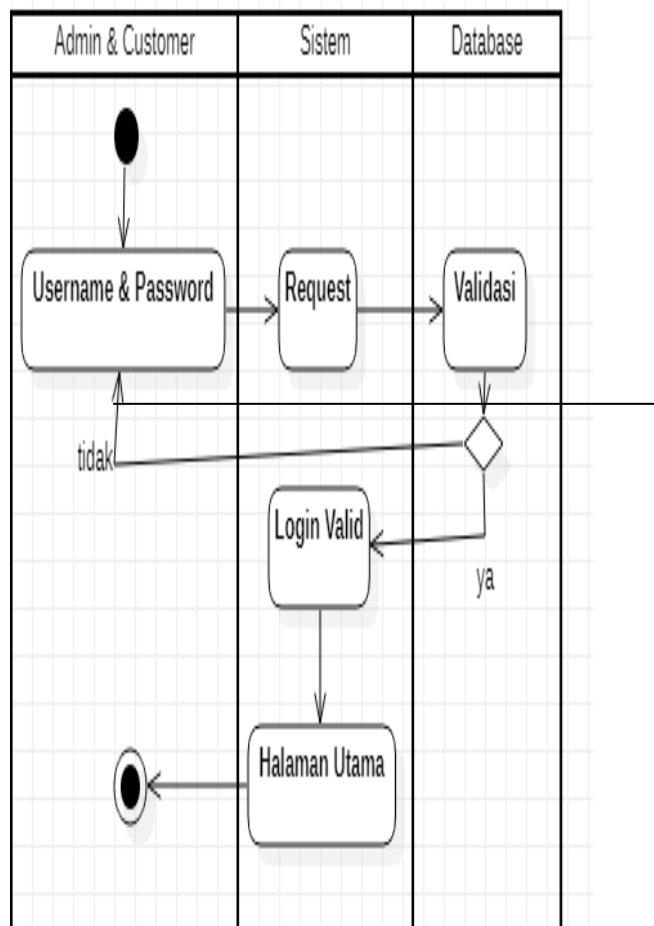
### 3.4.4 Activity Diagram

Activity diagram menggambarkan *workflow* suatu sistem atau proses bisnis. Diagram ini menunjukkan suatu alur kegiatan secara berurutan. Diagram ini digunakan untuk mendiskripsikan kegiatan-kegiatan dalam sebuah operasi meskipun juga dapat digunakan untuk mendeskripsikan alur kegiatan yang lainnya.

#### 3.4.4.1 Activity Diagram Login

User menginputkan *username & password* untuk *login*, kemudian sistem akan mengvalidasi *username & password* yang telah diinputkan. Sistem akan mengecek benar tidaknya *username & password* tersebut. Jika benar kemudian sistem akan masuk kehalaman utama, namun jika salah akan kembali kehalaman

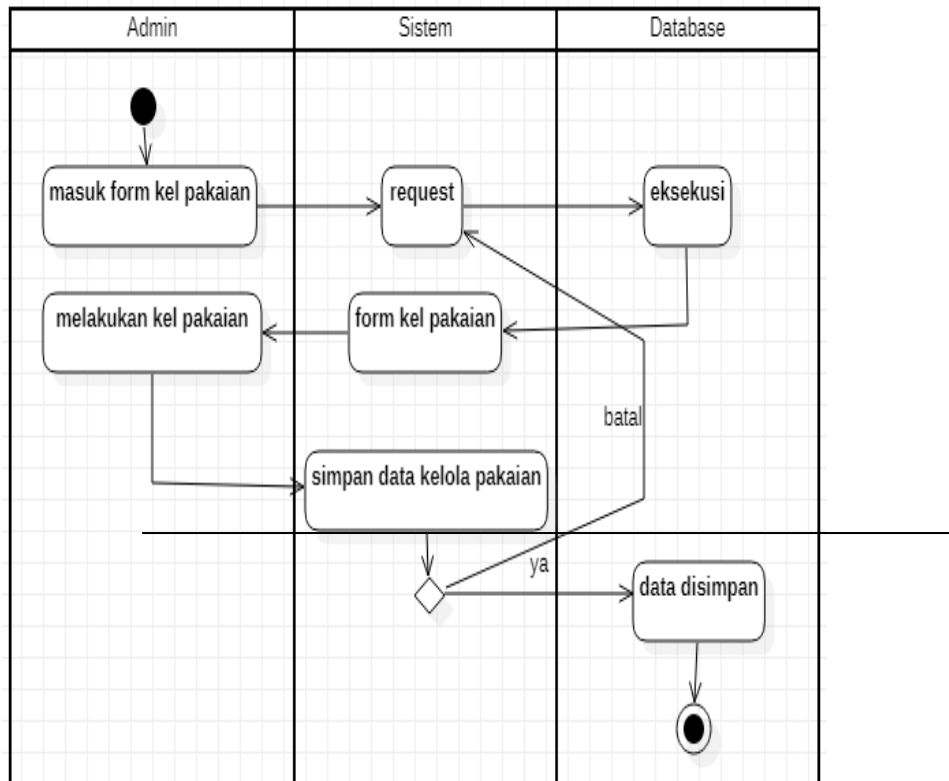
*login.*



Gambar 3.20 Activity diagram login.

#### 3.4.4.2 Activity kelola pakaian

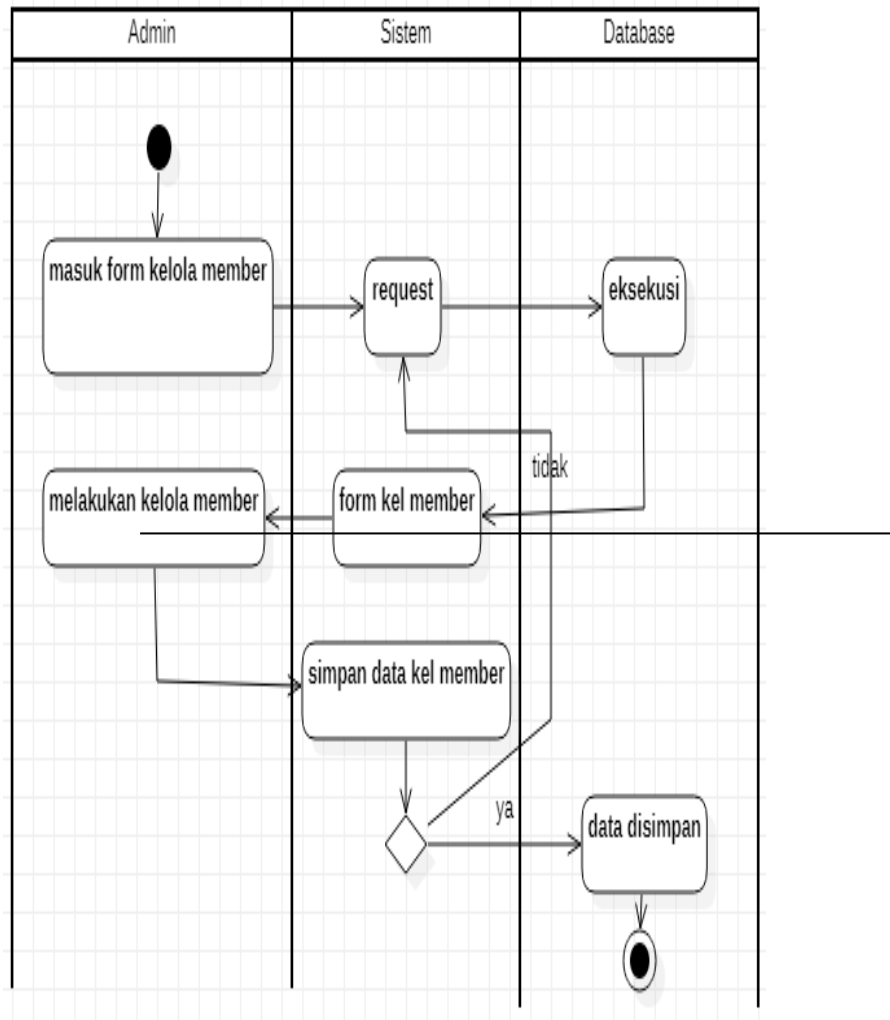
*Admin* masuk kehalaman kelola barang, untuk melakukan *input* barang dan admin memilih menu *input* berang, kemudian memasukkan data barang, setelah selesai menginputkan data barang, data akan disimpan.



Gambar 3.21 Activity kelola pakaian.

#### 3.4.4.3 Activity kelola member

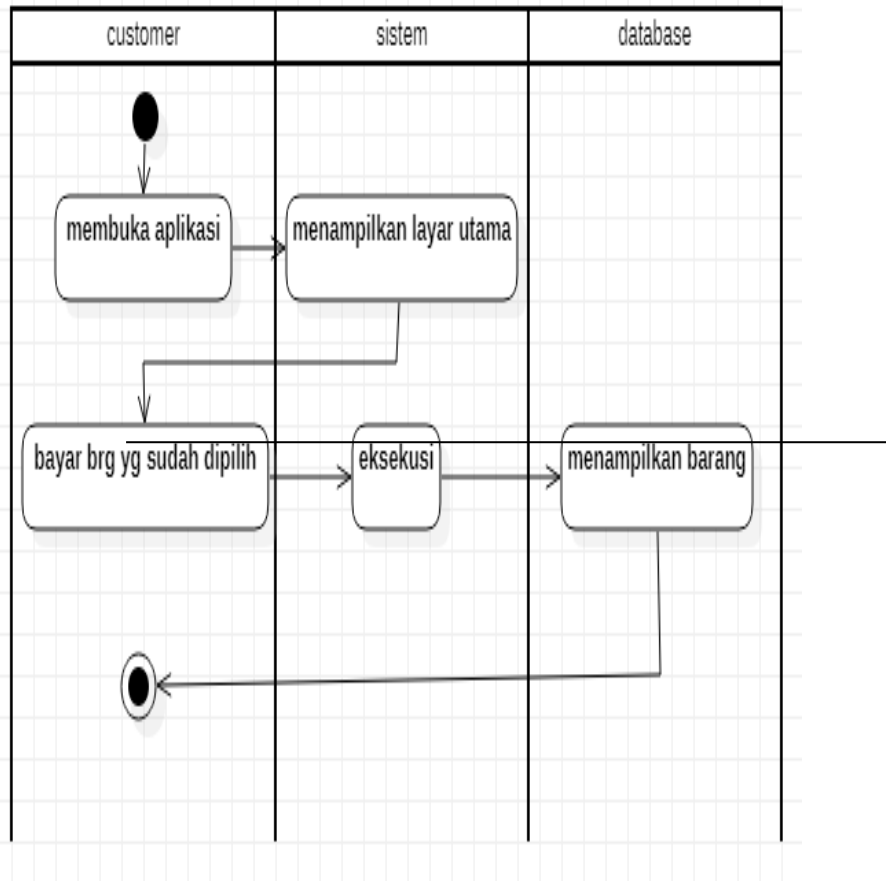
*Admin* akan masuk kedalam *kelola member*, untuk melakukan *input member admin* memilih menu *input member*, kemudian memasukkan data *member*, setelah selesai menginputkan data *member*, data kan disimpan.



Gambar 3.22 Activity kelola member.

#### 3.4.4.4 Activity Transaksi

*Customer* mencari barang yang akan dibeli, lalu menampilkan data barang, mengecek barang yang akan dibeli, lalu *transaksi* dan selesai.

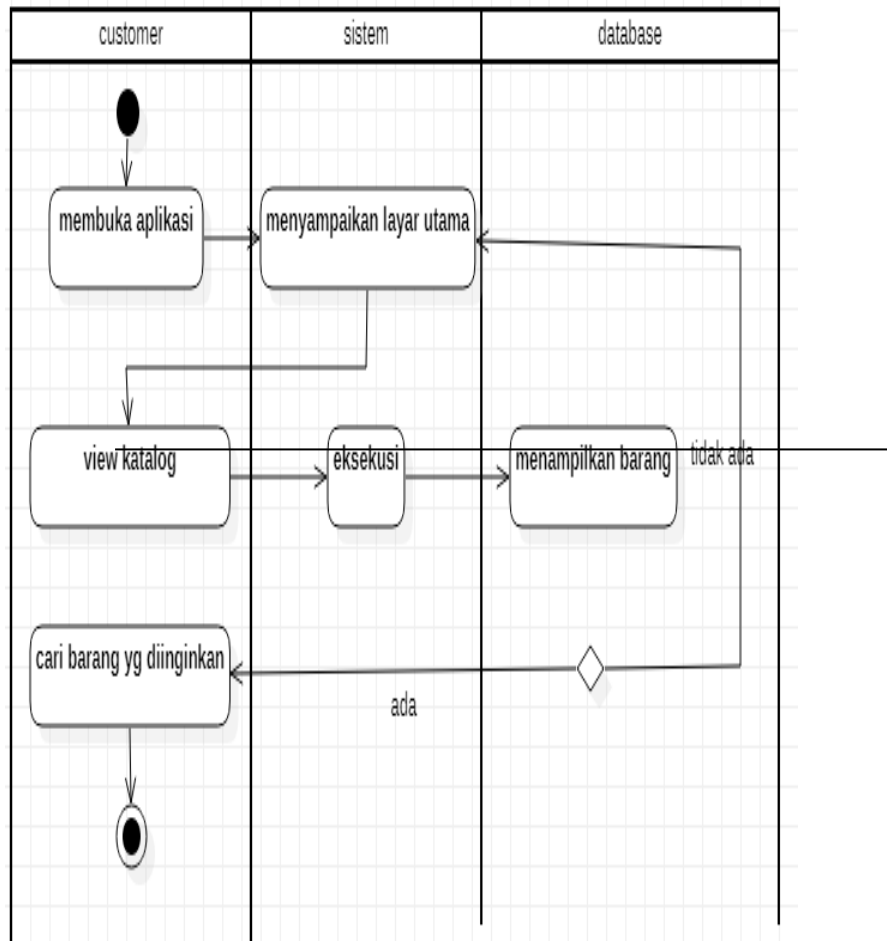


Gambar 3.23 Activity transaksi.

#### 3.4.4.5 Activity View Katalog.

*Customer* membuka aplikasi, lalu sistem menampilkan layar utama, dan *customer* menekan tombol *search*, cari barang dan sistem mengeksekusi, lalu database mencari barang, jika barang tidak ditemukan maka kembali ke layar utama, jika berhasil menampilkan barang yang diinginkan maka selesai.

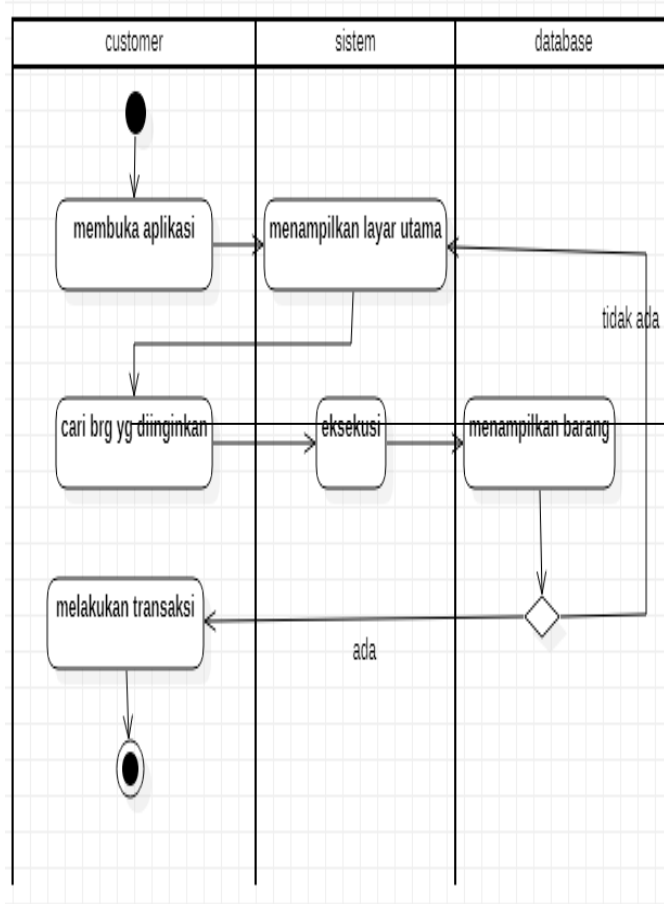




Gambar 3.24 Activity View Katalog.

#### 3.4.4.6 Activity Kernal Belanjaan

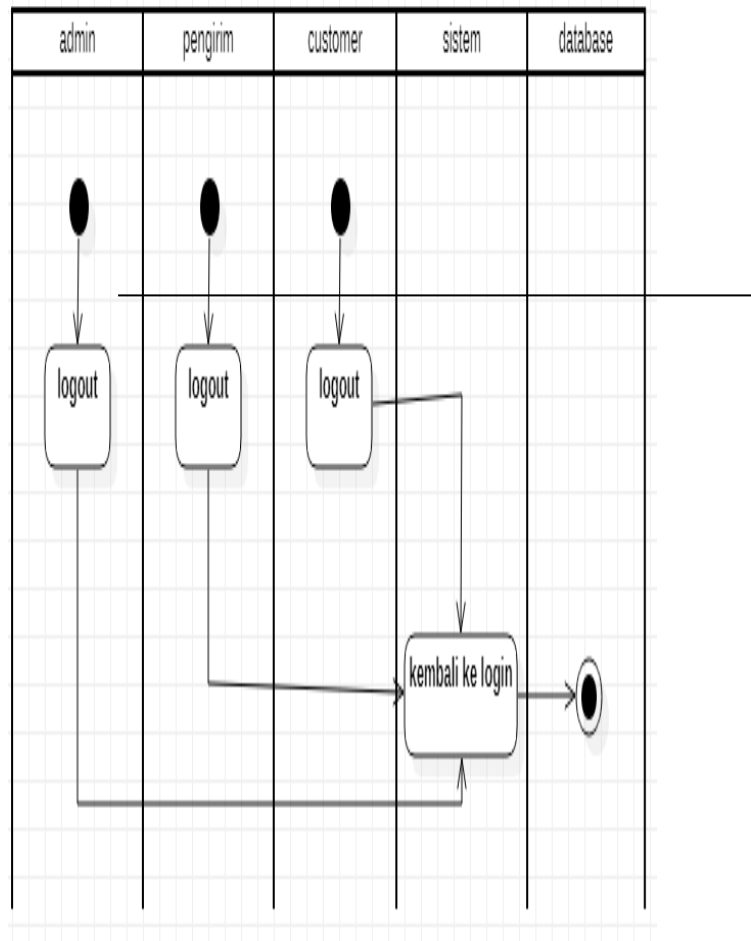
Customer membuka aplikasi, lalu system menampilkan layer utama, dan customer mencari barang yang diinginkan, dan system mengeksekusi, database menampilkan barang, jika barangnya ada maka customer melakukan pembayaran, dan jika tidak maka balik lagi ke layer utama.



Gambar 3.25 Activity Keranjang Belanjaan.

#### 3.4.4.7 Activity Logout

Admin, pengirim, dan customer meng*logout* aplikasi dan system akan menampilkan halaman *login*



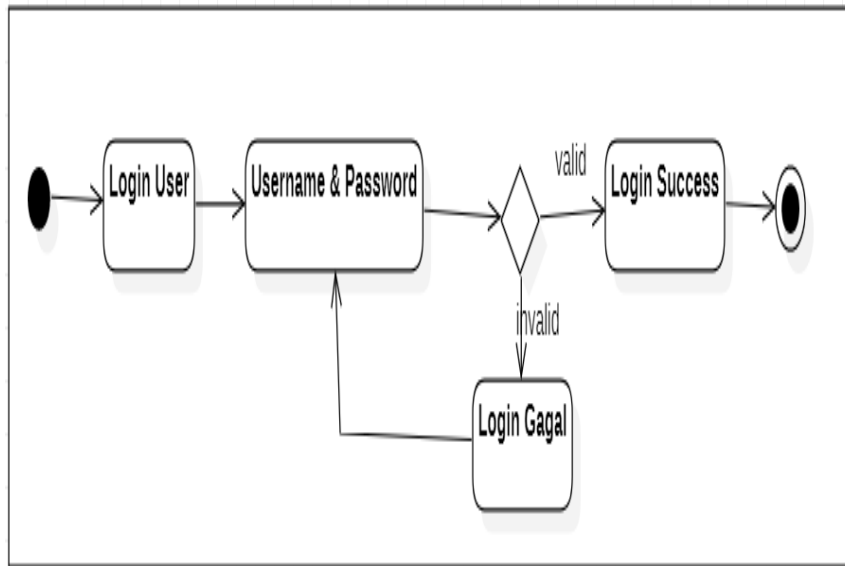
Gambar 3.26 Activity Logout.

### 3.4.5 Statechart Diagram

Diagram status digunakan untuk menyatakan kondisi sebuah objek pada saat sistem informasi berjalan. Diagram ini diadopsi dari penggambaran kondisi mesin status yang menggambarkan status apa saja yang dialami oleh mesin.

#### 3.4.5.1 Statechart Diagram Login

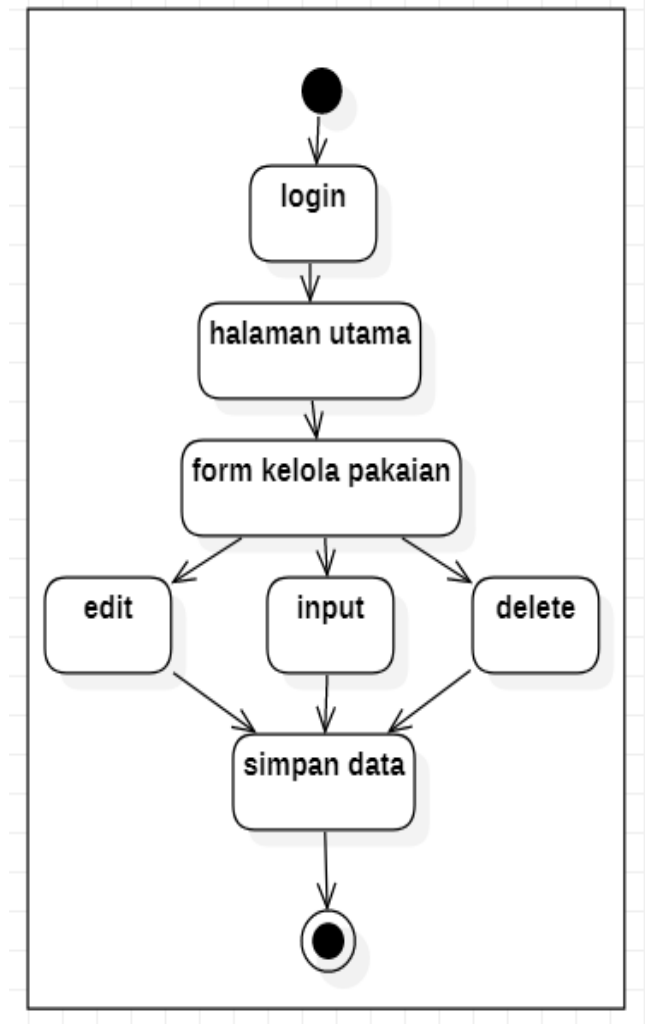
User memasukkan *username* & *password* untuk *login*, kemudian sistem akan mengvalidasi *username* & *password* untuk *login*, kemudian sistem akan mengvalidasi *username* & *password* yang telah diinputkan.



Gambar 3.27 Statechart Diagram Login.

#### 3.4.5.2 StateChart Kelola Pakaian

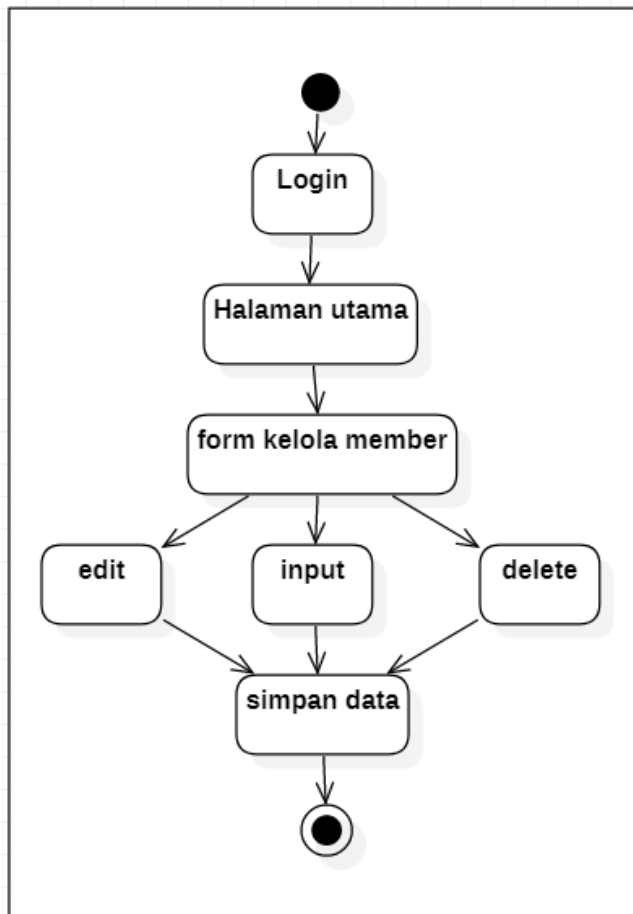
*Customer* mencari barang yang akan dibeli, lalu menampilkan data barang, mengecek barang yang akan dibeli, lalu *transaksi* dan selesai.



Gambar 3.28 *Statechart* Kelola Pakaian.

### 3.4.5.3 *StateChart* Kelola Member

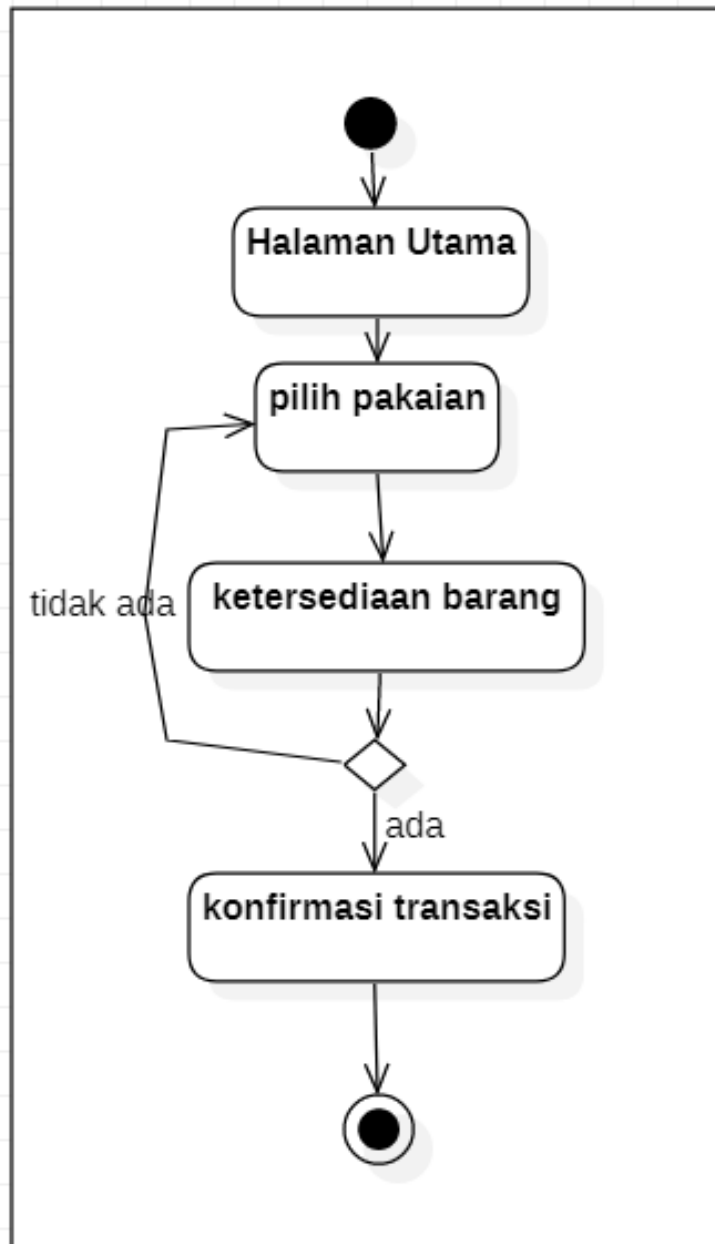
*Admin* akan masuk kedalam *kelola member*, untuk melakukan *input member* *admin* memilih menu *input member*, kemudian memasukkan data *member*, setelah selesai menginputkan data *member*, data kan disimpan.



Gambar 3.29 *Statechart Kelola Member.*

#### **3.4.5.4 Statechart Transaksi**

*Customer* mencari barang yang akan dibeli, lalu menampilkan data barang, mengecek barang yang akan dibeli, lalu *transaksi* dan selesai.

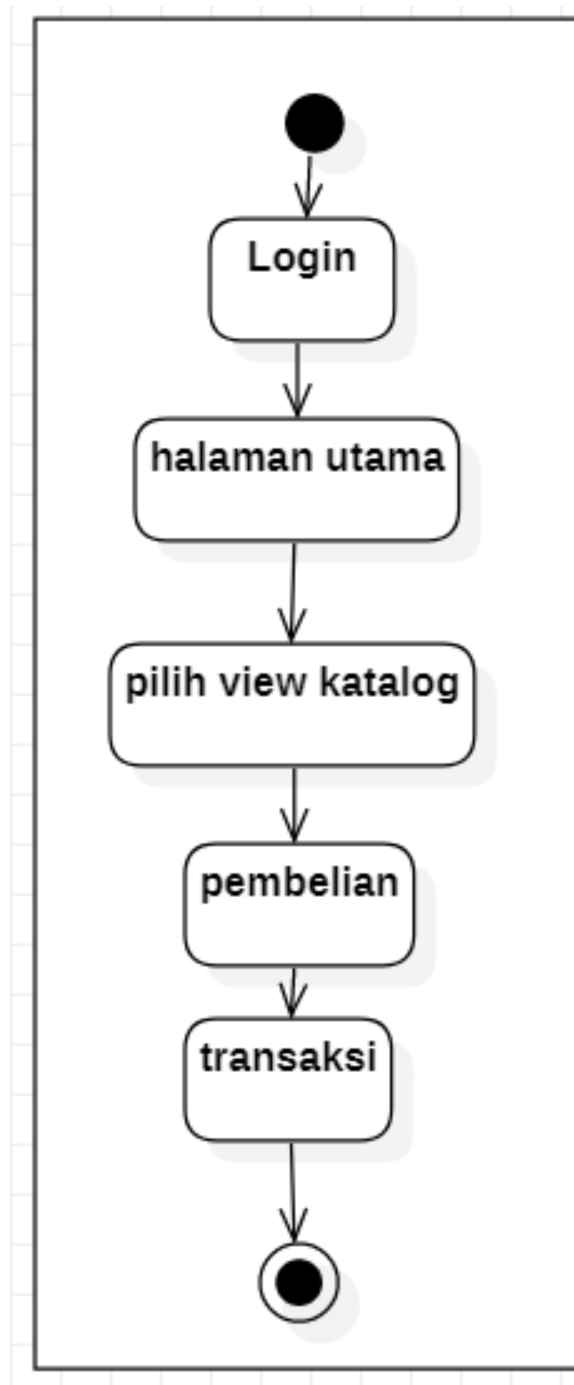


Gambar 3.30 *Statechart Transaksi.*

#### **3.4.5.5 Statechart View Katalog**

*Customer* membuka aplikasi, lalu sistem menampilkan layar utama, dan pilih *view catalog*, cari barang dan sistem mengeksekusi, lalu database mencari barang, jika barang tidak ditemukan maka kembali ke layar utama, jika

berhasil menampilkan barang yang diinginkan maka selesai.

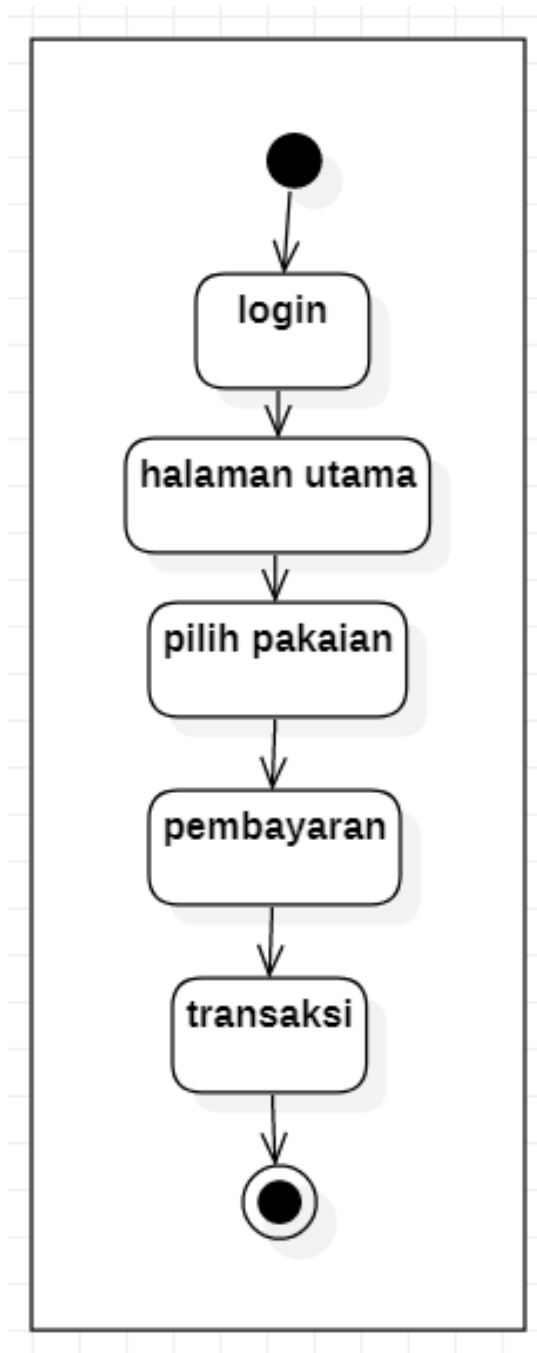


Gambar 3.31 *Statechart View Katalog.*

#### **3.4.5.6 Statechart Keranjang Belanjaan.**



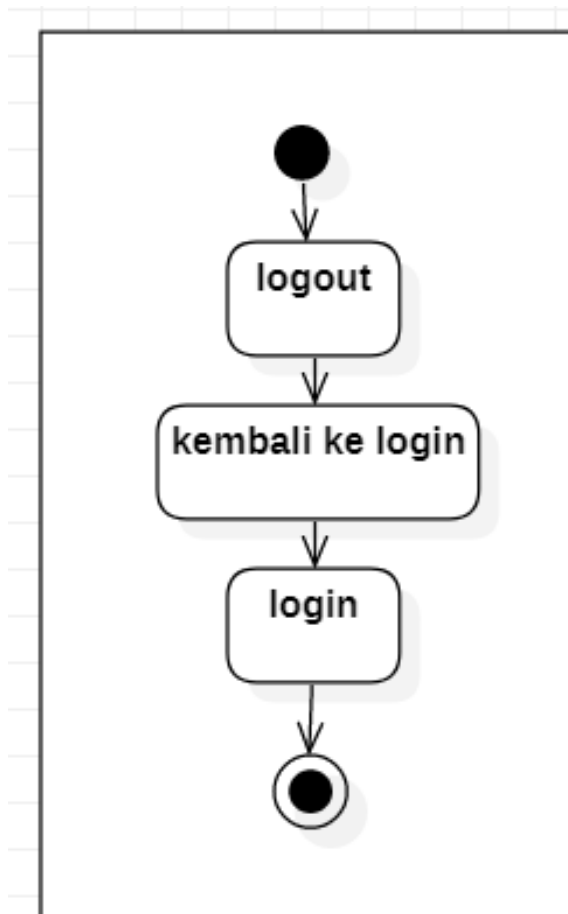
Login, kehalaman utama, lalu customer memilih pakaian, dan melakukan pembelian lalu pembayaran.



Gambar 3.32 *Statechart* Keanjang Belanjaan.

#### 3.4.5.7 Statechart Logout.

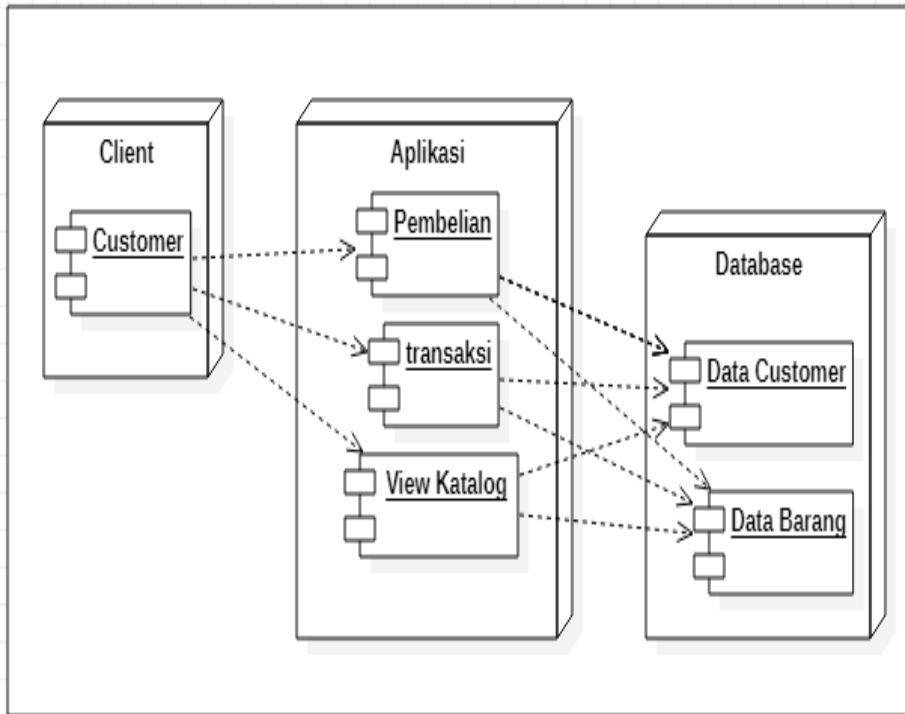
Jika ingin *logout*, lalu system akan menampilkan menu *login*



Gambar 3.33 Statechart Logout.

#### 3.4.6 Component Diagram

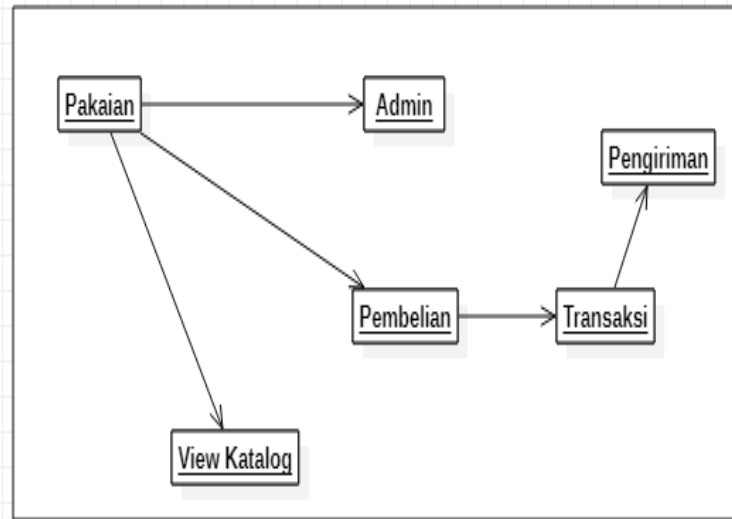
*Component Diagram* digunakan untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem.



Gambar 3.34 *Component Diagram*.

### 3.4.7 *Object Diagram*

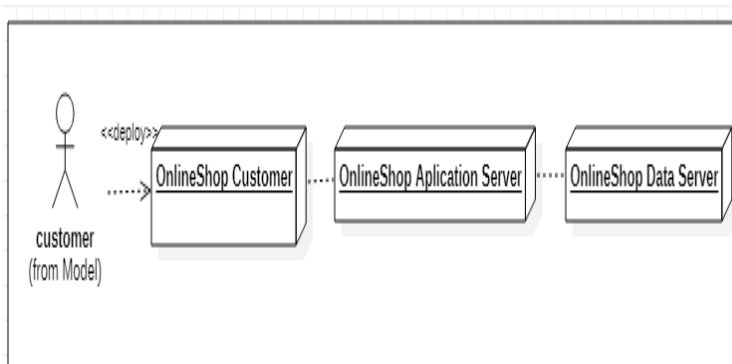
*Object Diagram* merupakan sebuah gambaran tentang objek-objek dalam sebuah sistem pada satu titik waktu. Karena lebih menonjolkan perintah-perintah daripada *class*, *object diagram* lebih sering disebut sebagai sebuah *diagram perintah*.



Gambar 3.35 *Object Diagram*.

### 3.4.8 *Deployment Diagram*

*Deployment Diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. *Diagram deployment* juga dapat digunakan untuk memodelkan hal-hal seperti sistem tambahan yang menggambarkan rancangan *device*, *node*, dan *hardware*, sistem *client/server*, sistem terdistribusi murni, dan rekayasa ulang aplikasi.



Gambar 3.36 *Deployment Diagram*

## Bab 4

---

## **Implementasi dan Pengujian**

---

### **4.1 Teori**

#### **4.1.1 Implementasi**

Implementasi sistem dilakukan ketika kebijakan telah dibuat, pekerjaan selanjutnya adalah mengatur, melakukan kepemimpinan untuk proses kepemimpinan dan melaksanakan proses kontrol.

Kegiatan terperinci dalam mengelola implementasi kebijakan dapat diatur melalui:

- Implementasi strategi
- Pengorganisasian
- Penggerak dengan baik
- Kontrol

Implementasi yang melibatkan upaya pembuat kebijakan untuk mengubah apa yang disebut "birokrat tingkat jalan" oleh Lipsky untuk menyediakan layanan sesuai dengan kebijakan kelompok sasaran. Untuk kebijakan sederhana, implementasi gratis yang melibatkan badan yang mengimplementasikan sebagai implementor, misalnya, kebijakan pengembangan infrastruktur publik untuk membantu orang-orang yang memiliki kehidupan yang baik, untuk keberhasilan implementasi kebijakan akan ditentukan dengan jumlah variabel atau aspek, dan masing-masing variabel tersebut dibahas secara setara satu sama lain. Kegiatan implementasi kebijakan sejak awal implementasi strategi, organisasi, gerakan kepemimpinan dan pengawasan dilakukan sesuai dengan apa yang diinginkan. Bab ini merupakan implementasi dari “Aplikasi Pembelian Pakaian Berbasis

Website” Implementasi merupakan tahap menerapkan hasil analisis dan perancangan ke tahap pembuatan program sesuai rancangan yang dilakukan.

#### **4.2.1 Prosedur Implementasi**

Dan pengorganisasian berikut merumuskan prosedur pelaksanaan, yang disusun dalam tipe dasar

- Desain organisasi dan struktur organisasi.
- Divisi pekerjaan dan desain pekerjaan.
- Integrasi dan koordinasi.
- Rekrutmen dan penempatan sumber daya manusia.
- Hak, otoritas juga kewajiban.
- Delegasi.
- Pengembangan kapabilitas organisasi dan kapabilitas sumber daya manusia.
- Budaya organisasi.
- Aspek berikutnya adalah mobilisasi dan kepemimpinan yang mengalokasikan sumber daya, mengelola prosedur implementasi dengan sumber daya.

Dalam fase kebijakan ini, ada dasar untuk fleksibilitas atau ruang bagi pelaksana individu untuk memilih tindakan otonom mereka sendiri dalam batas kewenangan

Dengan beberapa rencana:

- Efektivitas kepemimpinan.
- Motivasi.
- Norma.
- Kualitas.

- Kerja tim.
- Komunikasi organisasional.
- Perundingan
- Aspek selanjutnya adalah membahas proses dengan melakukan pemantauan berkala dengan beberapa rencana:
- Desain kontrol.
- Sistem manajemen informasi.
- Pemantauan
- Kontrol biaya atau keuangan.
- Audit.
- Implementasi kebijakan pada prinsipnya adalah langkah kebijakan yang dapat diambil oleh seseorang untuk mengambil keputusan.

### Implementasi Kebijakan Umum

Untuk menerapkan kebijakan umum, ada dua langkah yang tersedia, yang dilaksanakan langsung dalam bentuk program atau melalui perumusan kebijakan turunan atau berasal dari kebijakan umum, Implementasi adalah tindakan untuk mencapai tujuan yang diuraikan dalam ketentuan kebijakan, tindakan itu dilakukan baik oleh individu, baik pejabat pemerintah swasta. Sehubungan dengan uraian ini dapat ditarik kesimpulan jika implementasinya merupakan proses yang dinamis, di mana pelaksana kebijakan mengatur kegiatan atau kegiatan, hingga akhirnya hasil yang sesuai dengan maksud atau tujuan kebijakan akan diperoleh.

Menurut Hanifah Harsono (2002: 67)

Menurut Hanifah Harsono dalam bukunya yang berjudul Implementasi Kebijakan dan Kebijakan Politik mengenai implementasi atau proses sebagai berikut: Implementasi adalah proses implementasi kebijakan menjadi kebijakan kebijakan dalam mengembangkan kebijakan administrasi dalam rangka

meningkatkan program.

### **Definisi Implementasi Umum**

Gagasan di atas menunjukkan apakah implementasi kata bermuara pada kegiatan, perilaku, tindakan, atau sistem. Arti pemindahan ekspresi berarti jika diterapkan pada pelaksanaan suatu kegiatan, tetapi suatu kegiatan yang direncanakan dan dilaksanakan dengan perhatian penuh pada referensi ke norma-norma tertentu untuk mencapai tujuan kegiatan. Oleh karena itu, implementasi tidak berdiri sendiri tetapi dipengaruhi oleh objek berikutnya, kurikulum. Sesuai dengan kesepakatan, implementasi kurikulum Fullan adalah proses koordinasi, pemisahan program atau kegiatan dengan harapan orang lain dapat menerima dan melakukan perubahan. Dalam proses penyelesaian, evaluasi yang disebutkan di atas memberikan tekanan pada proses. Inti dari implementasi adalah proses, kegiatan yang digunakan untuk mentransfer / ide, program atau harapan sebagaimana diuraikan dalam kurikulum desain (disediakan) yang disetujui sesuai dengan desain. Masing-masing demi mencerminkan tingkat proses yang berbeda. Dalam diskusi dengan perjanjian, Nurdin dan Usman (2004) menjelaskan bahwa jika ditanya sebelumnya, evaluasi pelaksanaan dilakukan sebelum distribusi (diseminasi) desain kurikulum.

Proses program terkait dengan kegiatan yang berkaitan dengan menjelaskan tujuan program, menjelaskan sumber-sumber baru dan menunjukkan bagaimana menemukan apa yang digunakan. Keputusan kedua, menurut Nurdin dan Usman (2002), memprioritaskan fase perbaikan. Kata proses dalam dukungan menekankan hubungan antara pengembang dan guru (memfasilitasi pendidikan). Pengembang melakukan kontrol atas program yang baru disetujui, sumber baru, dan menggabungkan konten / materi program baru dengan sumber yang ada berdasarkan hasil uji coba lapangan dan pengalaman para guru. Hubungan antara pengembang dan guru terjadi dalam konteks penyempurnaan program, pengembang menyelenggarakan lokakarya atau diskusi



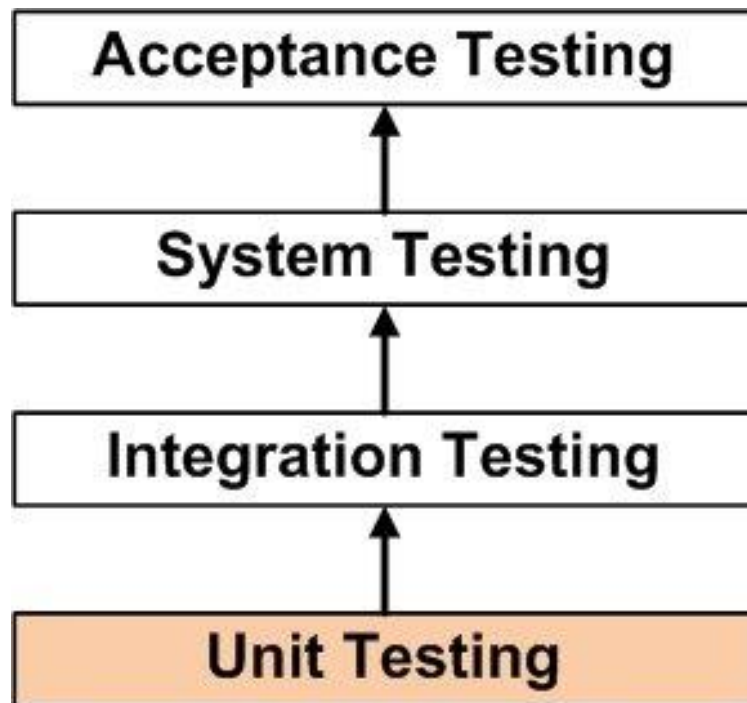
dengan guru untuk mendapatkan masukan. Implementasi dianggap selesai ketika proses peningkatan program baru tampaknya telah selesai. Sementara menyetujui tiga, Nurdin dan Usman (2002) melihat implementasi sebagai sisi dari program kurikulum. Implementasi dalam Kamus Besar Bahasa Indonesia disimpulkan sebagai suatu proses atau aplikasi. Berharap bahwa apa yang dilakukan dengan baik diimplementasikan dalam kurikulum yang telah dirancang / dirancang untuk diimplementasikan sepenuhnya.

#### **4.4 Analogi Implementasi**

Ketika misalkan dengan desain bangunan yang dibuat oleh insinyur bangunan tentang desain perumahan di kertas kalkir. Sehingga implementasi yang dilakukan oleh beberapa pengrajin yang telah dirancang telah dibuat dan tidak mungkin dilakukan atau direncanakan akan terdistorsi atau tidak sesuai dengan desain. Jika apa yang dilakukan oleh beberapa pabrikan tidak sama dengan hasil desain yang akan terjadi pada bangunan yang telah dibuat karena hasil dari proses yang panjang, rumit, sulit dan sempurna dalam hal desainer dan desain. Implementasi kurikulum juga diperlukan untuk melakukan apa yang telah dilakukan dalam kurikulum untuk dilakukan dengan sepenuh hati dan semangat. Masalah besar akan terjadi yang dilakukan dengan menentang atau menyimpang dari apa yang telah dibuat. Akan ada pemborosan antara desain dan implementasi.

##### **4.1.1 Lingkungan Implementasi**

Lingkungan implementasi dari suatu program aplikasi adalah perangkat keras dan perangkat lunak serta program aplikasi lainnya yang mendukung implementasi program aplikasi tersebut. Agar perancangan program aplikasi dapat berjalan dengan baik maka dibutuhkan perangkat keras pendukung dan perangkat lunak yang sesuai.



#### **4.1.2 pengujian**

##### **4.1.2.1 Proses pengujian**

Sistem berfungsi dengan baik dan mencari masalah yang mungkin terjadi sistem.

Tujuan pengujian adalah untuk bereksperimen:

1. Kesalahan bahasa (kesalahan bahasa), kesalahan yang disebabkan oleh jawab dalam pembaruan sintaksis.
2. Kesalahan waktu pemrosesan (kesalahan runtime), kesalahan kompilasi program dimulai. Kesalahan ini akan menyebabkan proses program berhenti sebelum berhasil berhenti.
3. Kesalahan logika, kesalahan yang disebabkan oleh logika program dibuat. Kesalahan ini sulit ditemukan karena tidak ada Ingat lokasi kesalahan.

##### **4.1.2.2 Rencana Tes**

Penilaian dilakukan dalam menguji penggunaan perangkat lunak ini mendiskusikan pengujian kotak hitam. Setuju dengan ini perangkat lunak fokus.

#### **4.1.2.3 Metode Pengujian**

Metode pengujian adalah metode atau teknik pengujian perangkat lunak, memiliki mekanisme untuk menentukan data uji yang dapat menguji perangkat lunak sepenuhnya dan memiliki kemungkinan tinggi menemukan kesalahan

Perangkat lunak dapat dirilis dalam dua cara, yaitu:

1. Pengujian menggunakan data uji untuk menguji semua elemen program (data internal, loop, logika, keputusan, dan jalur). Uji data dimunculkan dengan memahami struktur internal (kode sumber) dari perangkat lunak.
2. Pengujian dilakukan dengan mengeksekusi data uji dan memeriksa apakah Perangkat fungsional yang berfungsi dengan baik. Data uji dihasilkan dari spesifikasi perangkat lunak.

##### **1. UJI KOTAK PUTIH**

Pengujian kotak putih (kotak kaca) adalah tes yang didasarkan pada periksa detail desain, menggunakan struktur control program desain untuk pengujian procedural beberapa kasus pengujian. Penentuan masalah tes disesuaikan dengan struktur sistem, pengetahuan program yang digunakan untuk mengidentifikasi kasus uji tambahan.

Tujuan menggunakan kotak putih adalah untuk memprogram semua pernyataan.

Penggunaan metode pengujian kotak putih dilakukan untuk:

1. menyediakan modul
2. digunakan setidaknya sekali
3. gunakan semua keputusan logis untuk semua kondisi benar atau salah
4. mengeksekusi semua pengulangan nilai dan kendala operasional pada
5. setiap kondisi.
6. sengan digunakan struktur data internal untuk menjamin validitas jalur keputusan.

##### **A. Pengujian Jalur Dasar**


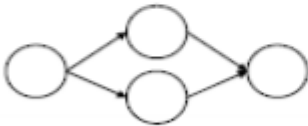
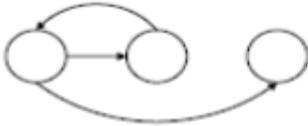

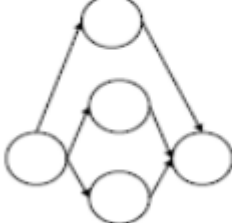
Tes jalur dasar adalah tes kotak putih pertama yang direkomendasikan oleh Tom

McCabe. Metode ini memungkinkan penguji untuk mengukur rumit seperti logis dari desain prosedural dan rumit panduan untuk membangun satu set dasar dari semua jalur eksekusi.

a. Notasi Diagram Alir

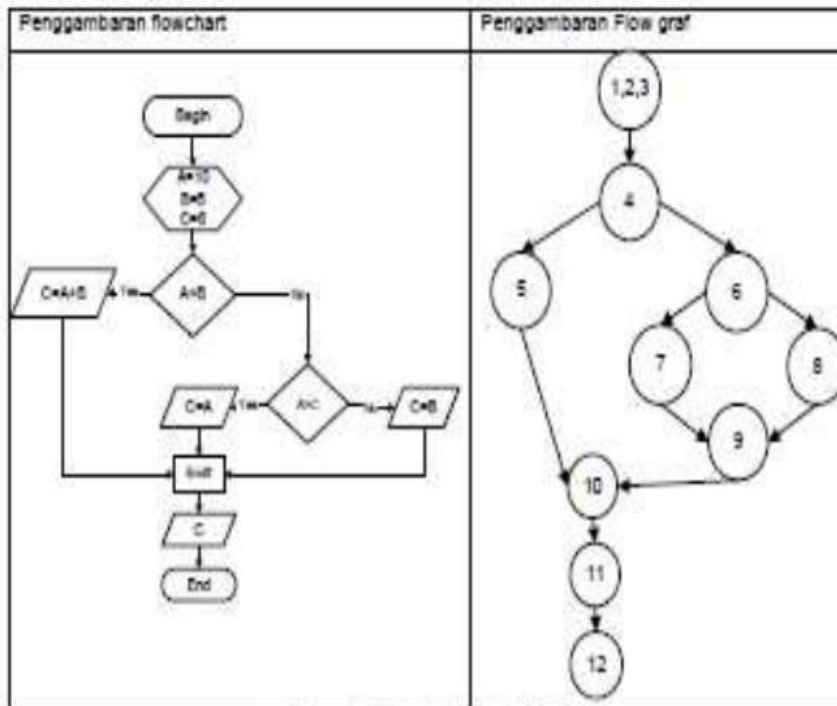
Notasi digunakan untuk menggambarkan jalur eksekusi notasi diagram alir (atau grafik program), yang menggunakan notasi loop atau simpul) dan panah. Notasi ini menggambarkan aliran kontrol logika yang digunakan dalam bahasa pemrograman.

**Tabel – Notasi Diagram Alir**

NOTASI	ARTI
	Langkah sequence
	Skema if
	Skema While (...) Do (...)
	Skema Repeat (...) Until (...)
	Skema Case (...) of

### Contoh Suatu PDL

```
Var
    A, B, C : integer
Begin
    A := 10;           (1)
    B := 5;            (2)
    C := 6;            (3)
    If A > B            (4)
        then C := A + B (5)
        Else if A > C   (6)
            then C := A (7)
            Else C := B; (8)
        Endif          (9)
    Endif              (10)
    Writeln('Nilai C = ', C); (11)
End.                  (12)
```



Gambar – Contoh Metode Basis Path

## B. Kompleksitas siklus

The Complexity Cycle adalah perangkat lunak metrik yang disediakan untuk pengukuran kuantitatif program logis, nilai yang diperoleh akan ditentukan oleh jumlah jalur independen di jalur yang ditentukan, juga memberikan nilai batas atas untuk jumlah tes yang harus dilakukan, untuk memastikan bahwa semua yang telah disepakati telah dieksekusi lihat satu kali. Jalur independen adalah jalur yang termasuk dalam program mengintegrasikan satu atau beberapa proses.

### Perhitungan siklus

1. Diagram alir diagram area (area wajah)
2.  $V(G) = E - N + 2$ .

E adalah jumlah tepi, dan N adalah jumlah node

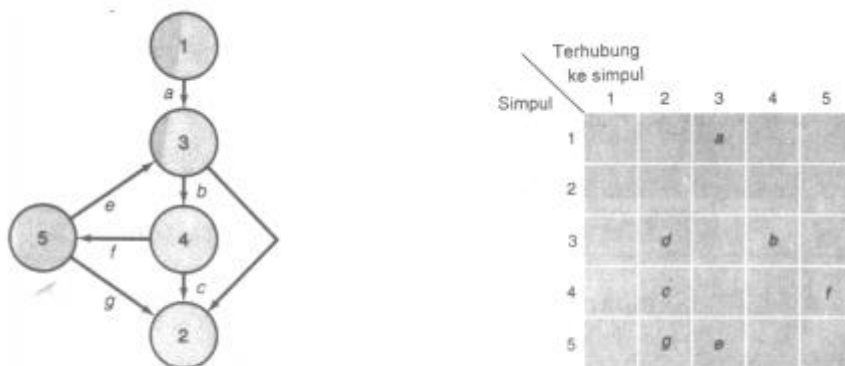
3.  $V(G) = P + 1$

P adalah simpul predikat Predicate Nodes adalah penggambaran simpul yang

memiliki satu atau lebih banyak input, dan lebih dari satu output.

### C. Graph Matrix (Graph Matrix)

Bentuk struktur data ini sering digunakan untuk pengujian adalah dengan matriks grafis. Matriks grafis adalah matriks persegi Jumlahnya sama dengan jumlah dalam bagan alur. Masukkan ke dalam matriks harus sesuai dengan arah. Matriks grafis selanjutnya disebut sebagai koneksi matriks, dan dijelaskan Sama dengan matriks tetangga dengan memperhatikan arah masuk dan keluar dari tepi.



Gambar – Contoh Matriks Koneksi

Dalam gambar flowgraph, setiap node ditandai dengan nomor sisi klan dengan huruf kecil, kemudian terjemahkan ke diagram matriks. Contoh hubungan antara simpul 3 dan simpul 4 pada grafik ditandai dengan huruf b. Timbangan bobot memberikan informasi tambahan tentang aliran kontrol. Dalam hubungan yang sederhana seseorang dapat dengan mudah diberi nilai 1 jika ada hubungan di antara mereka simpul atau nilai 0 jika tidak ada hubungan.

Terhubung ke simpul		1	2	3	4	5
Simpul						
1			1			
2						
3		1		1		
4		1				1
5		1	1			

**Gambar - Hubungan bobot**

Koneksi:

Simpul1:  $1-1=0$

Simpul2:  $2-1=1$

Simpul3:  $2-1=1$

Simpul4:  $2-1=1$

Cyclometric complexity= jumlah simpul +1

$$=3+4=4$$

### **Pengujian Struktur Kontrol**

Teknik pengujian jalur dasar adalah salah satu dari beberapa teknik untuk Memeriksa kontrol struktur, tetapi jalur pengujian dasar tidak memadai beberapa kasus uji.

#### **A. Kondisi Pengujian**

Ketentuan Penggunaan Pengujian program. Kondisi sederhana menggunakan variabel Boolean, dengan bentuk persamaan. E1 (operator-relasional) E2 Jika



suatu kondisi salah, maka akan ada komponen yang tidak ada dari kondisi yang salah, menghasilkan jenis kesalahan ketika kondisi berikut adalah:

1. operator kesalahan Boolean
2. Kesalahan variabel Boolean
3. tanda kurung Boolean Error
4. Kesalahan operator relasional
5. Kesalahan Persamaan Aritmatika

Contoh:

C1: B1 & B2

B1 & B2 adalah dua variabel Boolean sehingga mereka menjadi 22

B <sub>1</sub> = T ; B <sub>2</sub> = T	{(T,T), (T,F), (F,T), (F,F)}
B <sub>1</sub> = T ; B <sub>2</sub> = F	
B <sub>1</sub> = F ; B <sub>2</sub> = T	
B <sub>1</sub> = F ; B <sub>2</sub> = F	

#### b. Tes Aliran Data

Metode pengujian untuk menguji aliran data menggunakan resolusi variabel dalam program, efektif digunakan untuk melindungi kesalahan, Tetapi akan memiliki penilaian dan pemilihan jalur uji kompleks.m

#### c. Pengujian loop

Ada beberapa jenis loop, yaitu:

##### 1. Sederhana

Tes harus memperhatikan hal-hal berikut:

##### a) Mengabaikan seluruh loop

- b) Hanya ada satu jalan yang dilalui lingkaran
- c) Suatu variabel akan melewati loop jika nilainya lebih besar dari nilainya bertekad
- d) Harus memperhatikan batas variabel pada loop. (Kondisi:  $n-1$ ,  $n$ ,  $n + 1$ )

## 2. Nested Loop

Jumlah yang akan bertambah sesuai geometrik sesuai dengan jumlahnya lingkaran bersarang. Untuk menyederhanakan pengujian, itu harus pertimbangkan hal berikut:

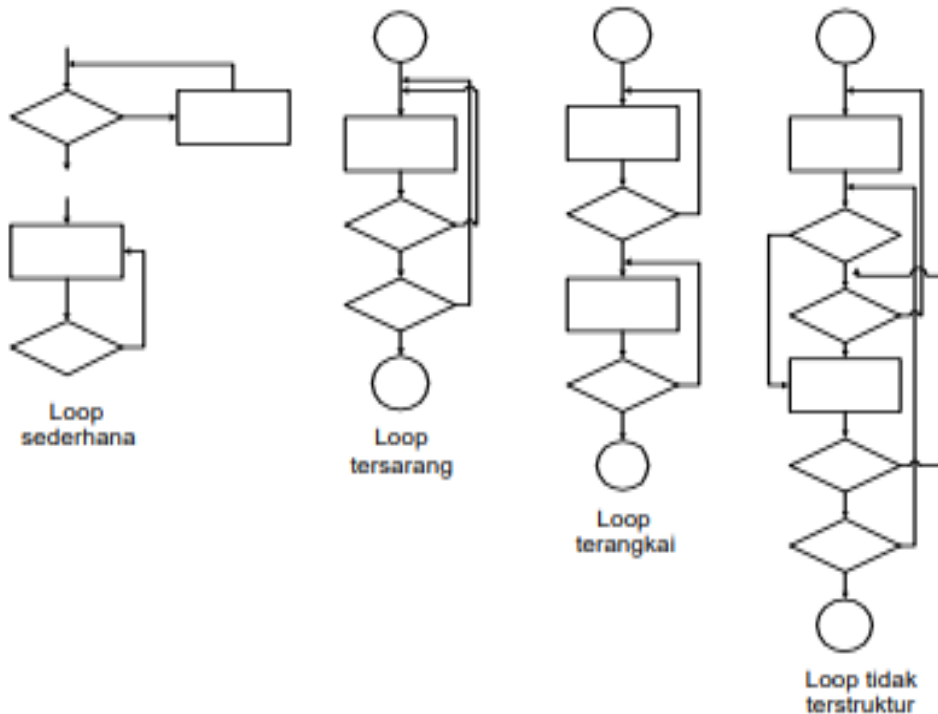
- a) memulai pengujian dari loop terdalam, dengan memulai pengujian dari minimumnya.
- b) lakukan tes loop sederhana dari loop terdalam ke loop terluar, dengan mempertimbangkan parameter yang digunakan.

## 3. Lingkaran Tegang

Tes untuk lingkaran harus disesuaikan sesuai dengan independensi variabel antara loop ini. Jika variabel digunakan di loop kedua tidak dihitung dengan loop pertama, maka pengujian loop digunakan sederhana, setiap kali lingkaran kedua bergantung pada nilai loop kedua, kemudian lakukan tes bersarang.

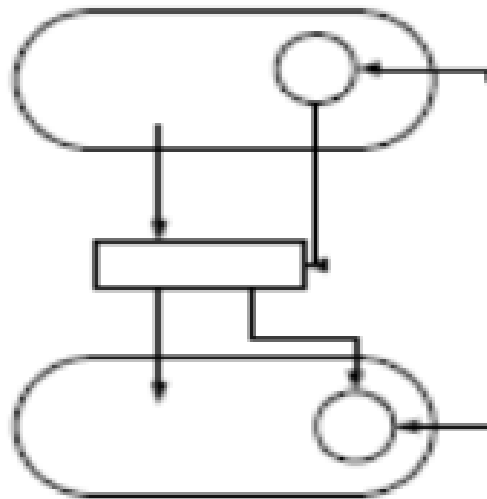
## 4. Loop Tidak Terstruktur

Sama sekali tidak direkomendasikan untuk digunakan dalam pembuatan program.



## B. MENGUJI KOTAK HITAM

Pengujian black box adalah pengujian pelengkap teknik putih kotak, karena pengujian kotak hitam diharapkan untuk mengungkapkan kelas kesalahan yang lebih luas daripada teknik kotak putih. Pengujian kotak hitam Sesuai dengan persyaratan masukan sesuai dengan persyaratan sesuai dengan persyaratan Program fungsional.



**Gambar – Black Box**

Pengujian black box adalah aspek mendasar dari pengujian sistem tanpa memperhatikan struktur logis internal perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak menggunakan dengan benar. Pengujian black box adalah metode desain data berbasis tes spesifikasi perangkat lunak. Uji data yang dihasilkan, jalankan di perangkat lunak dan kemudian output dari perangkat lunak diperiksa jika cocok seperti yang diharapkan.

Pengujian kotak hitam

1. fungsi yang salah atau hilang
2. Kesalahan antarmuka
4. kesalahan kinerja
5. kesalahan inisialisasi dan terminasi.

Berbeda dengan pengujian kotak putih, pengujian kotak hitam digunakan diterapkan di akhir ujian. Pengujian kotak hitam harus dimungkinkan jawab pertanyaan berikut:

- a. Apa validitas lisensi fungsional
- b. Kelas input apa yang akan membuat tes lebih baik




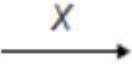


- c. Cara membatasi dari data yang terisolasi
- d. Kecepatan dan volume data apa yang akan ditoleransi system

Apa kombinasi data tertentu dengan sistem operasi.

### 3. Pengujian Berbasis Grafik

Langkah pertama dalam menguji kotak hitam adalah menguji suatu objek Ada dalam model perangkat lunak dan menentukan hubungan mana Dimaksudkan di antara benda-benda ini. Pengujian berdasarkan model grafis dilakukan pada perilaku sistem. Pengujian Grafis. Gambarkanlah grafik yang mewakili hubungan antara objek pada modul sehingga setiap objek dan dapat dipesan. Pengujian ini Dimulai dengan semua transisi, lalu jatuhkan lingkaran, di mana objek dan atribut diidentifikasi, juga memberikan indikasi titik awal dan berhenti.

Tabel – Notasi Pengujian berbasis graf

NOTASI	ARTI
	Simpul atau node Menggambarkan suatu objek
	Link Menggambarkan hubungan antar objek
	Node weight Menggambarkan properti atau nilai dari data
	Link weight Menggambarkan karakteristik link
	Link paralel Menggambarkan hubungan yang berbeda yang dibangun antar simpul
	Link simetris Menggambarkan hubungan dua arah antara dua objek

Ada beberapa pola berat tautan, yaitu:

- objek yang ditentukan

- Simetris, yaitu hubungan antara dua benda dalam dua arah
- Refleksif, yaitu hubungan yang mengarah ke simpul itu sendiri atau loop dibatalkan

Beizer membahas beberapa metode pengujian kotak hitam yang digunakan grafik, yaitu:

1. Transaction Flow Modeling, metode ini menggunakan simpul sebagai representasi langkah pada transaksi, dan tautan sebagai representasi hubungan logis antara langkah-langkah
2. Pemodelan keadaan terbatas, metode ini menggunakan node sebagai status dan tautan sebagai representasi. Statechart atau Grafik transfer status dapat digunakan untuk membuat grafik.
3. Pemodelan aliran data, metode ini menggunakan node sebagai repositori objek data dan tautan sebagai transformasi dari satu objek data ke objek data yang lain.
4. Pemodelan waktu, metode ini menggunakan simpul sebagai representasi objek program dan tautan sebagai hubungan berurutan antara objek.

#### C. Partisi Kesetaraan

Partisi ekivalensi adalah metode yang membagi domain input dari a program ke dalam kelas data, tentukan pengujian dengan pengujian mengungkap kelas kesalahan, sehingga akan mengurangi jumlahnya Kasus pengujian keseluruhan. Jika tautan memiliki pola transitivitas, simetris, dan refleksif akan ada kelas ekivalensi. Kelas ekivalensi mewakili Provisi Provisi valid dan tidak valid untuk kondisi input. Terutama, kondisi input dapat berupa harga numerik, kisaran harga, Harga terkait, atau kondisi Boolean.

##### Penentuan Kelas Kesetaraan

1. Ketika Input menentukan kisaran, maka kelas yang setara valid dan dua tidak valid
2. Jika input membutuhkan harga khusus, maka satu kelas valid dan dua yang

setara tidak valid

3. Jika suatu kondisi menentukan anggota mana yang ditentukan, maka satu kelas Satu atau dua padanan tidak valid
4. Jika kondisi input adalah Boolean, maka satu kelas valid dan satu ditentukan ditentukan.

Contoh:

Nomor telepon untuk SLJJ

Kondisi:

Kode area: kosong atau ada 3-4 digit dimulai dengan 0

Awalan: 3 digit tanpa awalan dengan 1 atau 0

Sufi: 4 digit

Kondisi input:

Kode area: Kondisi input Boolean - tidak ada yang bisa ada

Kondisi rentang input - nilai > 200 atau nilai < 999

Awalan: rentang rentang kondisi - nilai > 200 atau nilai < 999

Suffix: Kondisi harga input - 4 digit

#### B. Analisis Nilai Batas (Analisis Nilai Batas)

Analisis nilai batas adalah teknik desain proses yang melengkapi partisi kesetaraan, berdasarkan kesepakatan pada output domain. Pedoman untuk menentukan analisis nilai batas:

1. Jika input input adalah rentang yang ditentukan oleh nilai a dan b, pengujian harus dirancang dengan nilai a dan b, secara terus-menerus di atas dan di bawah ini dan sesuai
2. Jika input mengevaluasi jumlah input, maka uji harus dikembangkan menggunakan jumlah minimum dan Maksimum. Nilai tepat di atas dan di bawah minimum dan maksimum juga ditolak.
3. Pedoman 1 dan 2 juga berlaku untuk kondisi keluaran.
4. Jika struktur program data telah memerintahkan batas, maka Tes akan

dilakukan sesuai dengan ketentuan struktur data.

#### 2.1.2.3 Pengujian Komparatif

Tes yang diambil adalah metode pembuatan data yang dilakukan pada perangkat lunak yang dibuat berlebihan. Kekuatan Perangkat Lunak memiliki dua tim pengembangan, masing-masing berkembang Perangkat lunak individual untuk spesifikasi yang sama. Metode evaluasi digunakan untuk mengevaluasi data untuk salah satu perangkat lunak, yang kemudian digunakan sebagai input dalam tes perangkat lunak lain. Pengujian Komparatif digunakan untuk sistem yang menguji redundansi, Kasing uji dirancang untuk satu versi perangkat lunak yang dapat digunakan dalam menyetujui versi perangkat lunak lain, dan hasil dari versi kedua diharapkan perangkat lunaknya harus sama. Jika hasil pengujian kedua perangkat lunak ini berbeda maka kedua perangkat lunak akan digunakan untuk pencarian Shalat.

##### **4.1.1.1 Perangkat Lunak**

Kebutuhan perangkat lunak dalam system ini adalah:

- a. Sistem Operasi : Windows
- b. Pemrograman : Bootstrap

##### **4.1.1.2 Perangkat Keras**

Kebutuhan perangkat lunak dalam system ini adalah:

- a. Prosesor : Intel® Core i5-8250U CPU @ 1.60GHz
- b. Memori : 4 GB

#### **4.1.2 Pengujian**

### **4.2 Tampilan *User Interface***

#### **4.2.1 Tampilan Register**



Daftar Akun

Nama Anda

Username Anda

Password

Ulangi Password

Daftar

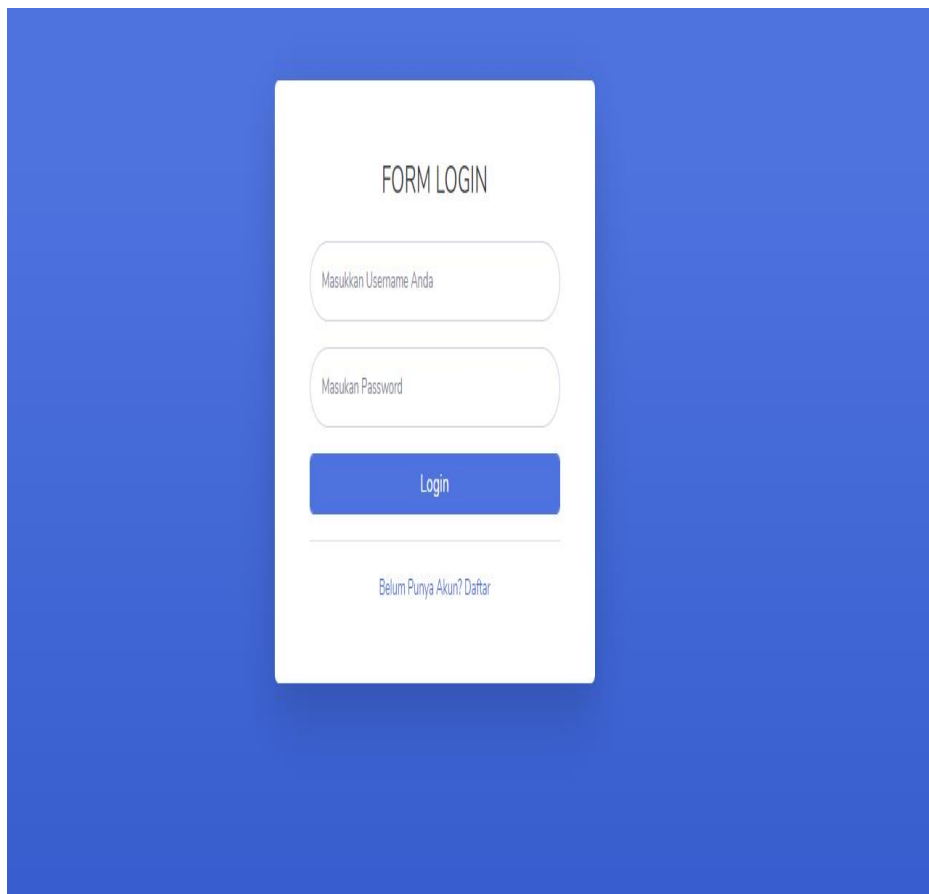
[Sudah Punya Akun? Silahkan Login](#)

Gambar 4.1 Tampilan *Register*

**Penjelasan:**

pada UI di atas adalah UI untuk register *customer* dimana ketika *customer* ingin melakukan pembelian atau pemesanan pakaian *customer* harus mendaftarkan diri agar bisa login dengan akun masing-masing yang telah diregister.

**4.2.2 Tampilan Login**



FORM LOGIN

Masukkan Username Anda

Masukkan Password

Login

[Belum Punya Akun? Daftar](#)

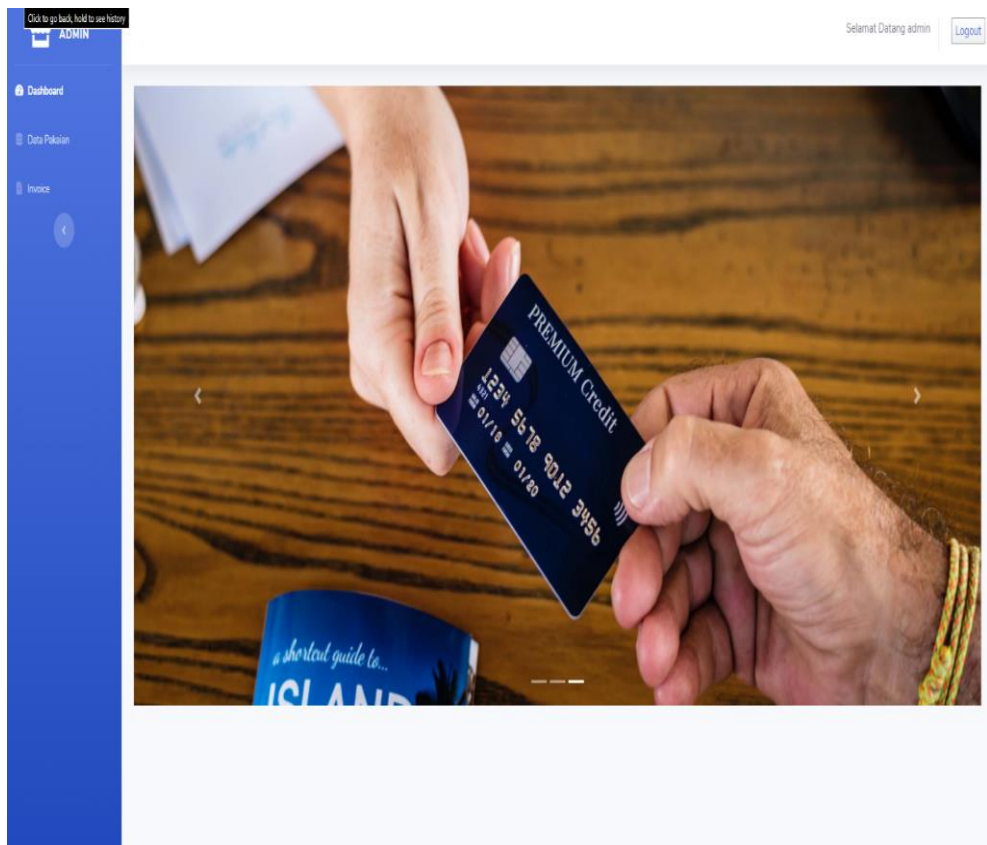
Gambar 4.2 Tampilan *Login*

**Penjelasan :**

pada tampilan di atas adalah tampilan untuk *login customer* dan *admin* dimana ketika *admin* dan *customer* ingin melakukan pembelian atau pemesanan pakaian *customer* dan *admin* harus login dengan akun masing-masing yang telah deregister.

**4.2.3 Tampilan Halaman Utama atau *Dashboard***

**4.2.3.1 Tampilan Halaman Utama *Dashboard Admin***

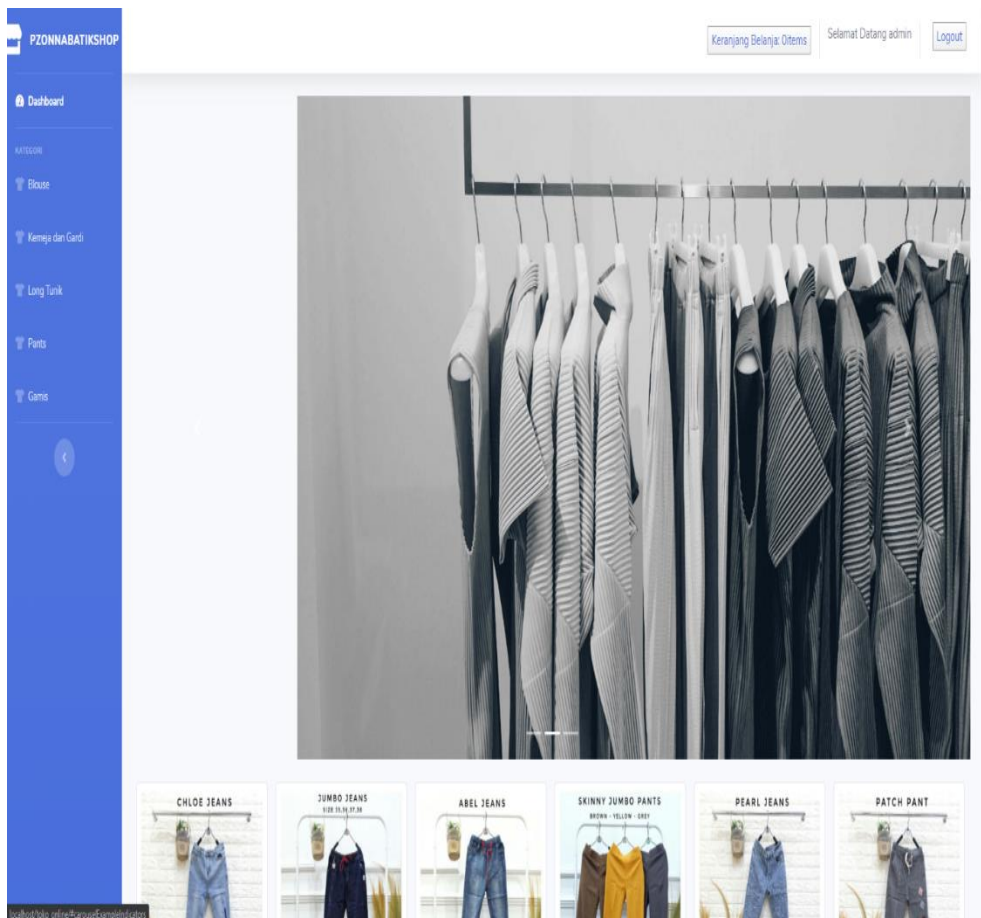


Gambar 4.3 Tampilan Halaman Utama *Dashboard Admin*

**Penjelasan:**

Pada halaman utama atau dashboard admin ini hanyalah tampilan biasanya untuk sambutan selamat datang atau tampilan memperindah saja yang isinya gambar-gambar slide

### 4.2.3.2 Tampilan Halaman Utama Dashboard Customer

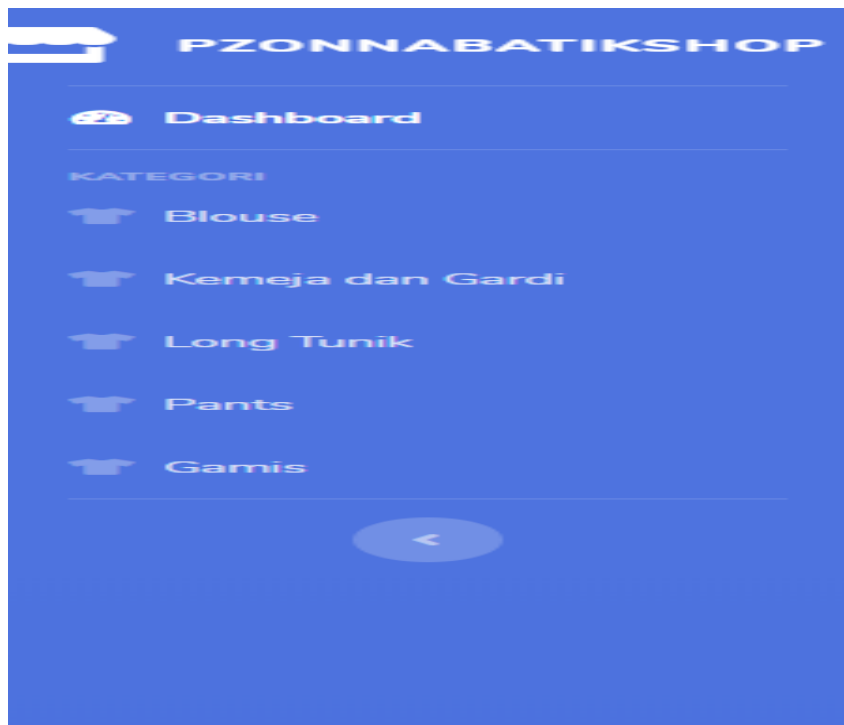


Gambar 4.4 Tampilan Halaman Utama *Dashboard Customer*

#### Penjelasan:

Tampilan diatas adalah gambaran dashboard untuk *customernya*, dimana isinya adalah produk-produk terbaru, dan promo, jadi *customer* bias melihat pada tampilan *dashboardnya*

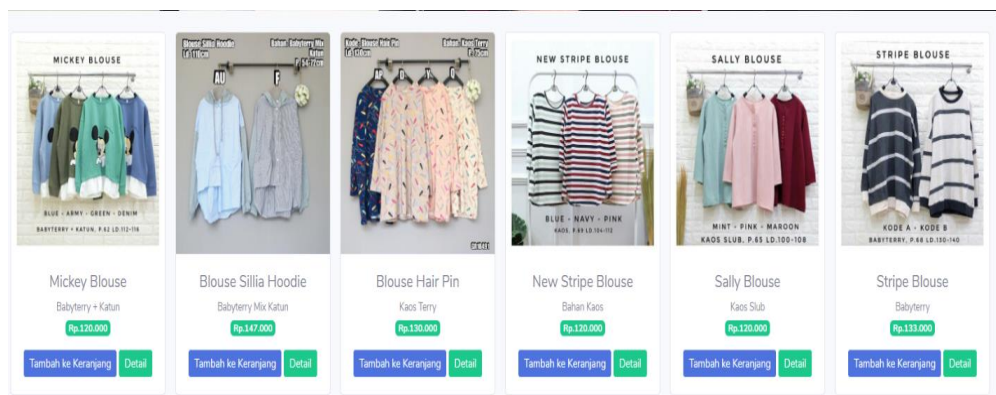
### 4.2.4 Tampilan kategori



Gambar 4.5 Tampilan kategori

### Penjelasan:


Tampilan di atas adalah tampilan untuk macam-macam kategori produk dimana ketika kita klik salah satu dari kategori diatas maka akan muncul pakaian dengan jenis kategori tersebut seperti misalnya kita klik kategori blouse maka akan tampil seperti dibawah ini



Gambar 4.6 Contoh Kategori

Dan pada button detail berisi detail pakaian seperti tampilan seperti dibawah ini:

Detail Pakaian



Nama Pakaian	Mickey Blouse
Keterangan	Babyterry + Katun
Kategori	Blouse
Stok	20
Harga	Rp.120.000

Tambah ke Keranjang

Kembali

Gambar 4.7 Detail Pakaian

Jika di klik kembali maka akan kembali ke tampilan halaman utama, Pada button tambah keranjang jika di klik akan masuk ke dalam keranjang belanja

#### 4.2.5 Tampilan Keranjang Belanja

Keranjang Belanja: Items

Selamat Datang fil

Logout

Keranjang Belanja

NO	Nama Produk	Jumlah	Harga	Sub-Total
1	Mickey Blouse	1	Rp. 120.000	Rp. 120.000
				Rp. 120.000

Hapus Keranjang

Lanjutkan Belanja

Pembayaran

Gambar 4.8 Tampilan Keranjang Belanja

#### Penjelasan:

Pada tampilan diatas adalah tampilan keranjang belanja, tampilan keranjang belanja memiliki 3 button hapus keranjang lanjutkan belanja dan pembayaran. Jika salah satu kita klik maka akan masuk ke dalam tampilannya masing-masing

#### 4.2.6 Tampilan Pembayaran

Gambar 4.9 Ta

Total Belanja Anda: Rp. 120.000

### Input Pengiriman dan Pembayaran

Nama Lengkap

Alamat Lengkap

No. Telepon

Jasa Pengiriman

Pilih BANK

Pesan






























mpilan Pembayaran

#### Penjelasan:

Pada tampilan diatas adalah tampilan pembayaran dimana kita bisa mengisi data lengkap kita dan jasa pengiriman yang kita inginkan, serta rekening bank yang harus kita transfer sesuai dengan pembelanjaan

#### 4.2.7 Tampilan Data Pakaian pada Admin



+ Tambah Pakaian								
NO	NAMA PAKAIAN	KETERANGAN	KATEGORI	HARGA	STOK	AKSI		
1	Chole Jeans	Size L, XL Jeans, P.85 LP.66 & PAHA.40-50 & 42-56	Pants	150000	19			
2	Jumbo Jeans	Jeans, P.89 LP.68-86 PAHA.54-62, Size 35,36,37,38	Pants	175000	20			
3	Abel Jeans	Jeans Import P.92 LP.68-80 PAHA.48-54 Size 31,32,33,34	Pants	160000	20			
4	Skinny Jumbo Pants	Katun Stretch, P.92 LP.76-92 PAHA.60-68	Pants	140000	20			
5	Pearl Jeans	Jeans Import P.93 LP.72-78 PAHA.46-50 Size L & XL	Pants	215000	20			
6	Patch Pant	Katun Stretch, P.88 L.68-74 PAHA.50-54 Size M-L (kecil)	Pants	155000	20			
7	Legging Ala Jeans	legging All Size	Pants	140000	20			
8	Button Pants	Wejes, P.90 LP.57-80 PAHA.60-68	Pants	140000	20			
9	Basic Jogger Pants	Katun, P.87 LP.60-72 PAHA.56	Pants	135000	20			
10	Jeans Skrit	Soft Jeans, P.92 LP.72	Pants	140000	20			
11	Legging Import	Kaos Import, P.90 LP.80-90 PAHA.60	Pants	130000	20			
12	Jogger Pants	Cotton Stretch LP.56-86 P.85	Pants	130000	20			
13	Gavril Pom Pom Pant	Katun Stretch, P.90 L.80-90 PAHA.64	Pants	125000	20			
14	Palla Isane Lumben	Isane Import P.92 LP.80-90 PAHA.60	Pants	175000	20			

Gambar 4.10 Tampilan Data Pakaian pada Admin

### Penjelasan:

Pada tampilan diatas adalah tampilan dimana kita bisa melihat data-data pakaian yang telah kita input, dan juga kita bisa mengedit datanya serta menghapus datanya sesuai dengan keinginan admin pada button tambah pakaian diatas disitulah tempat kita menampah data-data pakaian

#### **4.2.8 Tampilan Input Data Pakaian**

The image shows a web form titled "FORM INPUT PAKAIAN" with a close button (X) in the top right corner. The form contains the following fields:

- Nama Pakaian:** A text input field.
- Keterangan:** A text input field.
- Kategori:** A dropdown menu with "Blouse" selected.
- Harga:** A text input field.
- Stok:** A text input field.
- Gambar Pakaian:** A file upload field with a "Choose File" button and the text "No file chosen".

At the bottom right of the form are two buttons: a red "Close" button and a blue "Simpan" (Save) button.

Gambar 4.11 Tampilan Input Data Pakaian

### Penjelasan:

Tampilan diatas adalah tampilan dimana admin menginput data pakaian yang akan tampil pada data pakaian admin, dan kategori customer

### 4.2.9 Tampilan Invoice

Id Invoice	Nama Pemesan	Alamat Pengiriman	Tanggal Pemesanan	Batas Pembayaran	Aksi
28			2019-12-25 20:32:27	2019-12-26 20:32:27	<a href="#">Detail</a>
29	a	a	2020-01-09 14:53:57	2020-01-10 14:53:57	<a href="#">Detail</a>
30	dini	jl.sariasih	2020-01-10 09:53:01	2020-01-11 09:53:01	<a href="#">Detail</a>

Gambar 4.12 Tampilan Invoice

**Penjelasan:**

Tampilan di atas adalah tampilan untuk history bagi customer yang sudah membayar belanjaan dan detail pesanannya

---





































