BAB 3
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan. Analisis bertujuan untuk mendapatkan pemahaman secara keseluruhan tentang sistem yang akan dibuat berdasarkan masukan dari pihak-pihak dan juga pengalaman analis yang berkepentingan dengan sistem tersebut.

3.1.1 Analisis Masalah

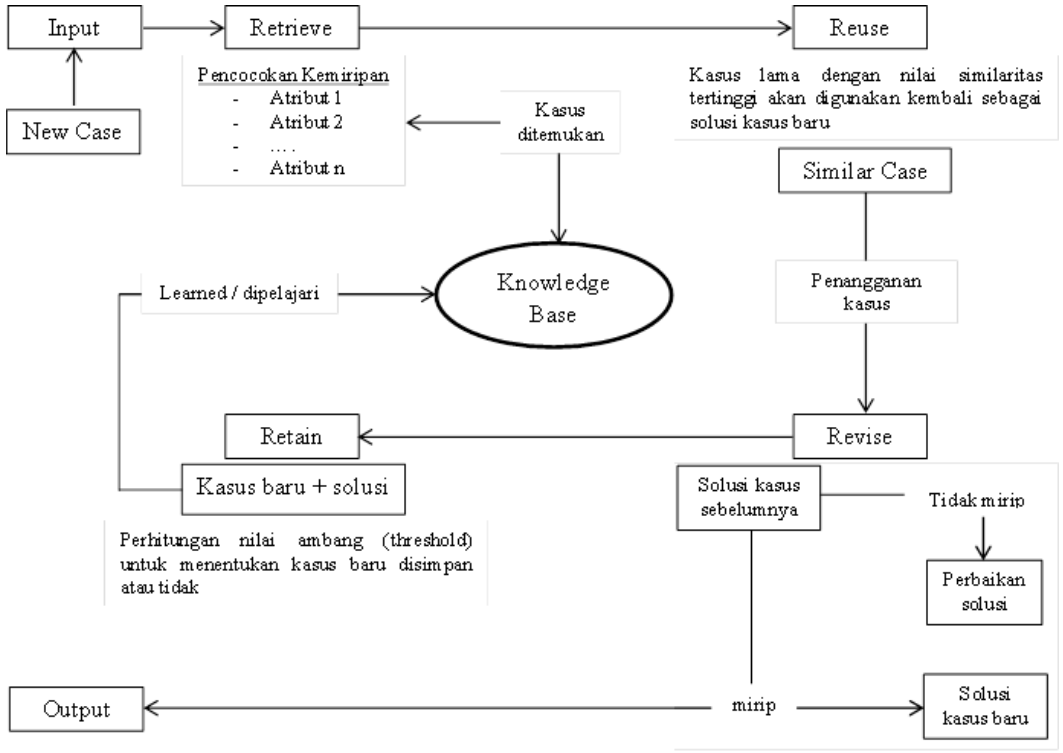
Analisis masalah menjelaskan tentang masalah apa saja yang ada sebelum dibangunnya aplikasi rekomendasi OOTD. Analisis masalah pada penelitian ini adalah sebagai berikut :

1. Responden kurang memiliki referensi pakaian untuk menghadari *event* tertentu.
2. Responden mengaku bahwa cuaca merupakan salah satu faktor yang memengaruhi dalam berpakaian.
3. Responden mengalami kebingungan dalam melakukan *mix and match* pakaian karena kurang mengetahui cara *mix and match item fashion* yang tepat.
4. Responden kurang memiliki referensi toko yang menyediakan *item fashion* yang mereka butuhkan.

3.1.2 Analisis Metode Rekomendasi OOTD (Outfit of The Day) Berdasarkan Event

Metode yang diterapkan untuk memberikan rekomendasi OOTD (*Outfit of The Day*) berdasarkan *event* adalah metode *Case-Based Reasoning* (CBR) yang menggunakan solusi masalah-masalah sebelumnya yang serupa. Pada prosesnya, metode CBR memiliki empat tahapan yaitu *retrieve, reuse, revise* dan *retain.* Salah satu komponen kunci CBR adalah kasus (*case*), satu atau lebih kasus akan

membentuk *case base* yang dijadikan basis pengetahuan (*knowledge base*) dalam proses CBR.



Gambar 3. 1 Tahapan CBR Pada Sistem

1. Retrieve

Retrieve adalah proses mengambil kasus-kasus lama dari *case base* yang mirip dengan kasus baru. Pada proses *retrieve* digunakan algoritma pencarian *nearest neighbour* yang memungkinkan untuk mencari kasus lama yang paling mirip dengan kasus baru. Hasil perhitungan dari algoritma *nearest neighbour* merupakan jarak kemiripan (*similarity*) kasus baru dan kasus lama. Berikut adalah rumus *similarity* untuk melakukan perhitungan algoritma *case-based reasoning* :

$$Similarity\ (problem, case) = \frac{(S1*W1)+(S2*W2)+(S3*W3)+(S4*W4)}{W1+W2+W3+W4}$$

Keterangan :

S1 : *similarity* (nilai kemiripan) jenis kelamin yaitu 1 (sama) dan 0 (beda)

S2 : *similarity* (nilai kemiripan) *event* yaitu 1 (sama) dan 0 (beda)

S3 : *similarity* (nilai kemiripan) tempat yaitu 1 (sama) dan 0 (beda)

S4 : *similarity* (nilai kemiripan) waktu yaitu 1 (sama) dan 0 (beda)

W1 : bobot jenis kelamin

W2 : bobot *event*

W3 : bobot tempat

W4 : bobot waktu

Pada pencarian *nearest neighbour*, kedekatan nilai-nilai atribut juga akan dihitung. Nilai kedekatan biasanya berada pada nilai 0 sampai dengan 1 dimana nilai 0 artinya kedua kasus tidak mirip sedangkan nilai 1 artinya kedua kasus mirip. Berikut adalah tabel data atribut dan nilainya.

Tabel 3. 1 Data Atribut dan Nilainya

No	Nama Atribut	Nilai Atribut
1	Jenis Kelamin	a. wanita b. pria
2	Event	a. Anniversaries b. Pernikahan c. Ulang tahun d. Acara sosial e. Reuni f. Liburan keluarga g. Perayaan budaya *h. Hangout* i. Kantor j. Olahraga k. Kampus l. Rekreasi m. Kenegaraan
3	Tempat	*a. Outdoor* b. *Indoor*
4	Waktu	a. Pagi b. Siang c. Sore d. Malam

a. Nilai Kedekatan Atribut Jenis Kelamin

Kedekatan nilai atribut jenis kelamin dapat dilihat pada tabel sebagai berikut:

Tabel 3. 2 Kedekatan Nilai Atribut Jenis Kelamin

Kedekatan	**wanita**	**pria**
wanita	1	0
pria	0	1

Nilai kedekatan didapat dari hasil penentuan antara kedekatan jenis kelamin pria dan wanita dimana dalam kasus ini laki-laki dan perempuan dinilai berbeda dalam hal menentukan OOTD (*Outfit of The Day*).

b. Nilai Kedekatan Atribut Event

Kedekatan nilai atribut *event* dapat dilihat pada tabel sebagai berikut :

Tabel 3. 3 Kedekatan Nilai Atribut Event

	A	B	C	D	E	F	G	H	I	J	K	L	M
A	1	0,9	0,7	0,3	0,4	0,2	0,3	0,1	0,1	0,3	0,1	0,2	0,1
B	0,9	1	0,7	0,5	0,6	0,5	0,5	0,2	0,1	0,1	0,1	0,5	0,1
C	0,7	0,7	1	0,3	0,6	0,7	0,8	0,9	0,1	0,1	0,1	0,6	0,1
D	0,3	0,5	0,3	1	0,7	0,5	0,8	0,3	0,6	0,2	0,5	0,2	0,5
E	0,4	0,6	0,6	0,7	1	0,5	0,6	0,4	0,1	0,1	0,1	0,4	0,1
F	0,2	0,5	0,7	0,5	0,5	1	0,7	0,8	0,1	0,1	0,1	0,7	0,1
G	0,3	0,5	0,8	0,8	0,6	0,7	1	0,8	0,6	0,2	0,5	0,1	0,5
H	0,1	0,2	0,9	0,3	0,4	0,8	0,8	1	0,1	0,1	0,1	0,8	0,1
I	0,1	0,1	0,1	0,6	0,1	0,1	0,6	0,1	1	0,6	0,8	0,1	0,8
J	0,3	0,1	0,1	0,2	0,1	0,1	0,2	0,1	0,6	1	0,8	0,1	0,8
K	0,1	0,1	0,1	0,5	0,1	0,1	0,5	0,1	0,8	0,8	1	0,1	0,8
L	0,2	0,5	0,6	0,2	0,4	0,7	0,1	0,8	0,1	0,1	0,1	1	0,1
M	0,1	0,1	0,1	0,5	0,1	0,1	0,5	0,1	0,8	0,8	0,8	0,1	1

Nilai kedekatan didapat dari hasil penentuan antara kedekatan jenis *event* yang satu dengan yang lainnya, dimana jenis *event* yang paling miriplah yang akan mempunyai nilai kedekatan yang cukup besar.

Keterangan :

A : *Anniversaries*
B : Pernikahan
C : Ulang Tahun
D : Acara Sosial
E : Reuni
F : Liburan Keluarga
G : Perayaan Budaya
H : *Hangout*
I : Kantor
J : Olahraga
K : Kampus
L : Rekreasi
M : Kenegaraan

c. Nilai Kedekatan Atribut Tempat

Kedekatan nilai atribut tempat dapat dilihat pada tabel sebagai berikut:

Tabel 3. 4 Kedekatan Nilai Atribut Tempat

Kedekatan	OD	ID
OD	1	0,5
ID	0,5	1

Nilai kedekatan didapat dari hasil penentuan antara kedekatan tempat yang satu dengan yang lainnya, dimana pada kasus ini jenis tempat dinilai berbeda.

Keterangan :

OD : *Outdoor*

ID : *Indoor*

d. Nilai Kedekatan Atribut Waktu

Kedekatan nilai atribut waktu dapat dilihat pada tabel sebagai berikut:

Tabel 3. 5 Kedekatan Nilai Atribut Waktu

Kedekatan	Pagi	Siang	Sore	Malam
Pagi	1	0,8	0,6	0,4
Siang	0,8	1	0,7	0,6
Sore	0,6	0,7	1	0,8
Malam	0,4	0,6	0,8	1

Nilai kedekatan didapat dari hasil penentuan antara kedekatan waktu yang satu dengan yang lainnya, dimana waktu yang paling miriplah yang akan memiliki nilai kedekatan yang cukup besar dan sebalinya.

Pada penelitian ini pembobotan atribut didapat dari hasil survey kepada 32 responden, besarnya bobot dari setiap atribut dapat dilihat pada tabel sebagai berikut :

Tabel 3. 6 Bobot Atribut

Nama Atribut	Bobot	Persentase (bobot x 100%)
Jenis kelamin	0,52	52 %
Event	0,27	27 %
Tempat	0,15	15 %
Waktu	0,06	6 %

Berikut ini adalah contoh kasus rekomendasi OOTD (*Outfit of The Day*) berdasarkan *event*, dimana bobot asal sudah ditentukan sebesar 60% dan pada *knowledge base* rekomendasi OOTD sudah terdapat 4 kasus yang dapat dilihat pada tabel berikut ini :

Tabel 3. 7 Knowledge Base Rekomendasi OOTD

No.	Id	Display_src	Event	Jenis Kelamin	Waktu	Tempat	Tag
1	1462856610472643690	https://scontent-sin6-2.cdninstagram.com/t51.2885-15/e35/17125351_1668892446740894_8243873705273327616_n.jpg	Kenegaraan	wanita	pagi	*indoors*	*blazer,pant suit,spring jacket,satchel,skirt suit,peacoat,cross body bag,capris,skirt,wide leg pants,cardigan,midi skirt,womens boots,leather jacket,knee length skirt,dress,tote bag,jean skirt,maxi skirt,mini skirt*
2	1535409587522374242	https://scontent-sit4-1.cdninstagram.com/t51.2885-15/e35/19121595_671013283099791_6115307940808228864_n.jpg	Hangout	wanita	siang	*indoors*	*capris,tube top,jeans,jumpsuit,bodysuit,pant suit,wide leg pants,overalls,denim jacket,kimonos,kimono,boyfriend pants,halter top,leather jacket,womens shorts,boot cut pants,jean skirt,spring jacket,prom dress,vest*
3	1455671742672687275	https://scontent-sin6-2.cdninstagram.com/t51.2885-15/e35/16906675_1324937904220810_6906573536731594752_n.jpg	Pernikahan	wanita	siang	*outdoors*	*maxi skirt,midi skirt,capris,sarong,dress,overalls,womens boots,wide leg pants,kimono,jumpsuit,womens sandals,skirt,cardigan,womens shorts,relaxed pants,robe,womens scarf,raincoats,button-down,womens hat*
4	1490035738635441344	https://scontent-sit4-1.cdninstagram.com/t51.2885-15/e35/17881180_16800665012298473_3380595112600928256_n.jpg	Pernikahan	pria	siang	*outdoors*	*capris,mens shorts,denim jacket,spring jacket,mens boots,sneakers,boat shoes,blazer,wide leg pants,mens sandals,overalls,jumpsuit,skinny pants,mens hat,platform shoes,womens boots,leather jacket,womens hat,peacoat,vest*

Tabel 3. 8 Contoh Kasus Baru

No	Id Kasus	Event	Jenis Kelamin	Tempat	Waktu
1	X	Ulang tahun	Pria	*Outdoors*	Siang

1. Perhitungan Kasus 1

Tabel 3. 9 Nilai Similarity Atribut Kasus X dan 1

Id Kasus : X	Similarity (X,1)	Id Kasus : 1
Wanita	0	Pria
Kenegaraan	0,1	Ulang tahun
Indoors	0,5	*Outdoors*
Pagi	0,8	Siang

Bobot rekomendasi *event* X :

Jenis kelamin : 0,52 Waktu : 0,15

Event : 0,27 Tempat : 0,06

Similarity (X,1) = $\frac{(S1*W1)+(S2*W2)+(S3*W3)+(S4*W4)}{W1+W2+W3+W4}$

$= \frac{(0*0,52)+(0,1*0,27)+(0,5*0,15)+(0,8*0,06)}{0,52+0,27+0,15+0,06}$

$= \frac{0+0,027+0,075+0,048}{1}$

$= \frac{0,15}{1} = 0,15$

2. Perhitungan Kasus 2

Tabel 3. 10 Nilai Similarity Atribut Kasus X dan 2

Id Kasus : X	Similarity (X,2)	Id Kasus : 2
Wanita	0	Pria
Hangout	0,9	Ulang tahun
Indoors	0,5	*Outdoors*
Siang	1	Siang

Bobot rekomendasi *event* X :

Jenis kelamin : 0,52 Waktu : 0,15

Event : 0,27 Tempat : 0,06

Similarity (X,2) = $\frac{(S1*W1)+(S2*W2)+(S3*W3)+(S4*W4)}{W1+W2+W3+W4}$

$= \frac{(0*0,52)+(0,9*0,27)+(0,5*0,15)+(1*0,06)}{0,52+0,27+0,15+0,06}$

$$= \frac{0 + 0{,}243 + 0{,}075 + 0{,}06}{1}$$

$$= \frac{0{,}378}{1} = 0{,}378$$

3. Perhitungan Kasus 3

Tabel 3. 11 Nilai Similarity Atribut Kasus X dan 3

Id Kasus : X	Similarity (X,3)	Id Kasus : 3
Wanita	0	Pria
Pernikahan	0,7	Ulang tahun
Outdoors	1	*Outdoors*
Siang	1	Siang

Bobot rekomendasi *event* X :

Jenis kelamin : 0,52 Waktu : 0,15

Event : 0,27 Tempat : 0,06

Similarity (X,2) $= \frac{(S1*W1)+(S2*W2)+(S3*W3)+(S4*W4)}{W1+W2+W3+W4}$

$$= \frac{(0 * 0{,}52)+(0{,}7 * 0{,}27)+(1 * 0{,}15)+(1 * 0{,}06)}{0{,}52+0{,}27+0{,}15+0{,}06}$$

$$= \frac{0 + 0{,}189 + 0{,}15 + 0{,}06}{1}$$

$$= \frac{0{,}399}{1} = 0{,}399$$

4. Perhitungan Kasus 4

Tabel 3. 12 Nilai Similarity Atribut Kasus X dan 4

Id Kasus : X	Similarity (X,4)	Id Kasus : 4
Pria	1	Pria
Pernikahan	0,7	Ulang tahun
Outdoors	1	*Outdoors*
Siang	1	Siang

Bobot rekomendasi *event* X :

Jenis kelamin : 0,52

Event : 0,27

Waktu : 0,15

Tempat : 0,06

$$Similarity\ (X,2) = \frac{(S1*W1)+(S2*W2)+(S3*W3)+(S4*W4)}{W1+W2+W3+W4}$$

$$= \frac{(1*0{,}52)+(0{,}7*0{,}27)+(1*0{,}15)+(1*0{,}06)}{0{,}52+0{,}27+0{,}15+0{,}06}$$

$$= \frac{0{,}52+0{,}189+0{,}15+0{,}06}{1}$$

$$= \frac{0{,}919}{1} = 0{,}919$$

2. Reuse

Berdasarkan hasil perhitungan pada tahap *retrieve*, diperoleh nilai-nilai kemiripan (*similarity*) setiap kasus yang ditunjukan pada tabel berikut ini :

Tabel 3. 13 Nilai Similarity Perbandingan Kasus

Id Kasus	Nilai Similarity dengan Kasus Baru (X)	Persentase (Nilai Similarity x 100%)
1	0,15	15 %
2	0,378	37,8 %
3	0,399	39,9 %
4	0,919	91,9 %

Nilai-nilai tersebut kemudian dibandingkan dengan jumlah nilai bobot dengan nilai bobot asal yaitu 0,60 atau dalam persentasenya 60%.

3. Revise

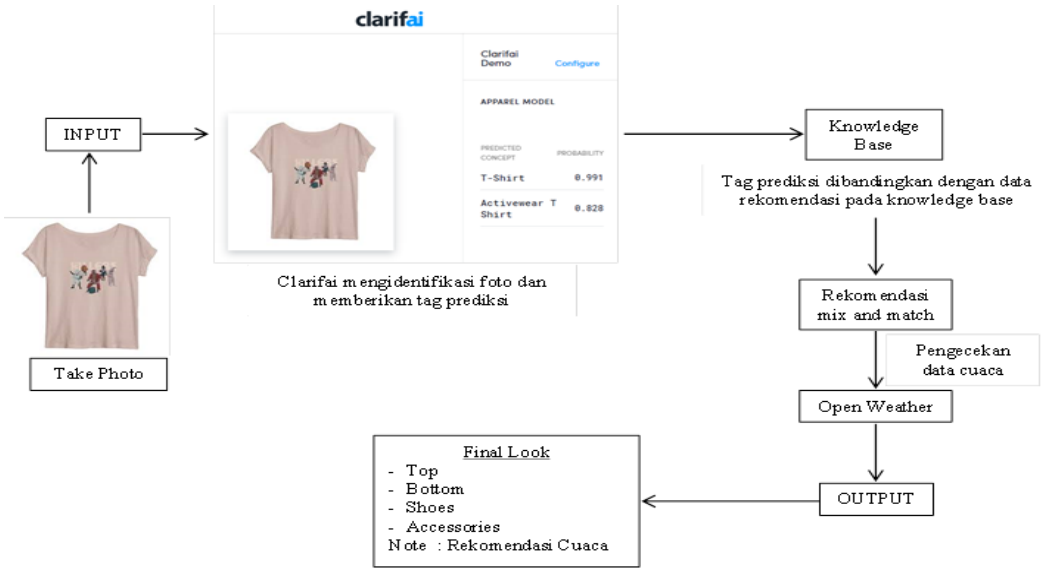
Pada contoh kasus pertama menghasilkan solusi tingkat kemiripan tertinngi 91,9%, jadi rekomendasi OOTD (*Outfit of The Day*) berdasarkan *event* yang dihasilkan dapat langsung diberikan. Akan tetapi, jika ternyata setelah dilakukan proses perhitungan dan menemukan kasus yang kemiripannya di bawah 60% maka dilkukan proses *revise.*

4. Retain

Pada tahapan ini sistem mulai menambah aturan dengan memasukkan data kasus baru yang sudah ditemukan solusinya ke dalam *knowledge base* yang nantinya dapat digunakan untuk kasus berikutnya yang memiliki permasalahan yang sama.

3.1.3 Analisis Metode Rekomendasi OOTD (Outfit of The Day) Berdasarkan Mix and Match Item

Pencarian solusi untuk rekomendasi OOTD (*Outfit of The Day*) berdasarkan *mix and match item* menggunakan Clarifai. Clarifai akan mengenali foto yang di-*upload* oleh *user* dan kemudian memberikan *tag* prediksi dari foto berdasarkan model prediksi yang dipilih. Prediksi yang dihasilkan akan menjadi data masukkan untuk pencarian solusi pada *knowledge base* OOTD. Setelah sistem memberikan rekomendasi yang cocok dengan *tag* prediksi Clarifai, sistem akan memberikan informasi data cuaca pada saat itu dan kemudian akan memberikan rekomendasi tambahan berdasarkan cuaca pada saat itu. Proses pencarian solusi untuk rekomendasi OOTD (*Outfit Of The Day*) berdasarkan *mix and match item* adalah sebagai berikut.



Gambar 3. 2 Proses Rekomendasi OOTD Berdasarkan Mix and Match Item

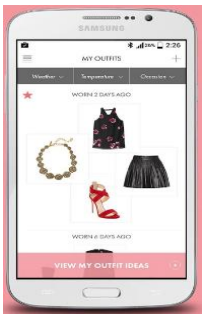
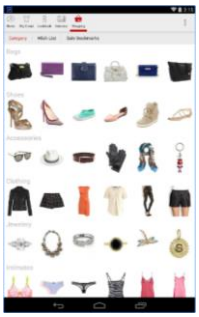
3.1.4 Analisis Aplikasi Sejenis

Analisis aplikasi sejenis dimaksudkan untuk menganalisis fungsionalitas dan alur dari aplikasi yang telah ada. Hal ini bertujuan untuk memilah fungsionalitas dan alur yang akan diadopsi pada sistem yang diteliti berdasarkan

manfaat yang sudah dihasilkan dari aplikasi yang dianalisis dan mencari *uniqueness* dari perangkat lunak yang akan dibangun pada penelitian ini.

Beberapa aplikasi yang memberikan informasi tentang OOTD (*Outfit of The Day*) adalah sebagai berikut:

Tabel 3. 14 Analisis Aplikasi Sejenis

No	Aplikasi	Kelebihan	Kekurangan
1	https://play.google.com/store/apps/details?id=net.catalystww.android.stylitics 	- Memiliki fitur *closet* yang bias menyimpan jenis pakaian, sepatu dan aksesoris - Memiliki fitur perancanan OOTD menggunakan kalender	- Tidak memiliki fitur rekomendasi OOTD - *Mix and match* OOTD dilakukan sendiri oleh *user*
2	https://play.google.com/store/apps/details?id=com.fabu.stylicious&hl=en 	- Memiliki fitur *closet organizer* - Memiliki fitur *style inspiration* yaitu tampilan *style* OOTD dari *fashion stylist*	- *Mix and match* OOTD dilakukan sendiri oleh *user*

3.1.5 Analisis Teknologi

Analisis teknologi yang terdapat dalam penelitian ini adalah sebagai berikut:

1. GPS

Global Positioning System (GPS) merupakan sebuah alat atau sistem yang dapat digunakan untuk menginformasikan penggunanya dimana dia berada

(secara global) dipermukaan bumi yang berbasis satelit. Data dikirim dari satelit berupa sinyal radio dengan data digital. Cara kerja GPS adalah sebagai berikut :

a. Menggunakan perhitungan *triangulation* dari satelit
b. Perhitungan *triangulation*, GPS mengukur jarak menggunakan *travel time* sinyal radio
c. Dalam mengukur *travel time*, GPS memerlukan akurasi waktu yang tinggi
d. Perhitungan jarak hanya bisa dilakukan apabila posisi pasti satelit dan ketinggian pada orbitnya diketahui
e. GPS harus mengoreksi *delay* waktu perjalanan di atmosfir sampai diterima *receiver.*

GPS *receiver* mengambil informasi dengan perhitungan *triangulation*, menghitung lokasi *user* dengan tepat dan kemudian membandingkan waktu sinyak yang dikirim dengan waktu sinyal tersebut diterima. Berdasarkan informasi itu dapat diketahui berapa jarak satelit, dengan diketahui jarak satelit GPS *receiver* dapat melakukan perhitungan dan menentukan posisi *user* dan menampilkan peta elektronik.

2. JSON

JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca, ditulis, serta mudah diterjemahkan dan dibuat oleh komputer. JSON merupakan format teks yang tidak bergantung pada bahasa pemograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh *programmer* dengan bahasa C, C++, C#, Java, JavaScript, Perl, Python dll. Struktur format JSON yang digunakan pada penelitian ini adalah sebagai berikut:

a. Data *Object*

Object adalah sepasang nama/nilai yang tidak berurutan. *Object* dimulai dengan { (kurung kurawal buka) dan dengan } (kurung kurawal tutup).

Tabel 3. 15 Struktur JSON Data Object

Struktur data objek

{
"key" : "value"
}

b. Data *Array*

Array adalah kumpulan nilai yang berurut. *Array* dimulai dengan [(kurung kotak buka) dan diakhiri dengan] (kurung kotak tutup).

Tabel 3. 16 Struktur JSON Data Array

Struktur data array

{
"result" : "value",
"item" : [{ "key" : "value" },{ "key" : "value" }]
}

c. Data *Null*

Data *null* digunakan ketika data yang diterima dai *server* merupakan data tunggal atau berupa satu objek.

Tabel 3. 17 Struktur JSON Data Null

Struktur data null

{
"result" : "value",
"message" : "value",
"status_code":value
}

3. RESTful

REST (*REoresentational State Transfer*) adalah model arsitektur yang pada dasarnya memanfaatkan teknologi dan protocol yang sudah ada seperti HTTP (*Hypertext Transfer Protocol*) dan XML. REST *web service* merupakan *web service* yang dibangun dengan memanfaatkan teknologi dan protokol yang sudah ada. Penggunaan metode-metode HTTP dalam REST adalah sebagai berikut:

Tabel 3. 18 Metode HTTP dan Penggunaannya dalam REST

Metode	Deskripsi
GET	Mendapatkan (*read*) sebuah sumber daya (*resource*) yang diidentifikasi dengan URI (*Uniform Resource Identifier*).
POST	Mengirimkan sumber daya (*resource*) ke *server*. Metode ini digunakan untuk membuat (*create*) sumber daya baru.
PUT	Mengirimkan sumber daya (*resource*) ke *server*. Digunakan untuk memasukkan (*insert*) atau memperbarui (*update*) sumber daya yang tersimpan.
DELETE	Menghapus (*delete*) sumber daya (*resource*) yang diidentifikasi dengan URI.
HEAD	Mendapatkan metadata (*response header*) dari sumber daya (*resource*) yang diidentifikasi dengan URI.

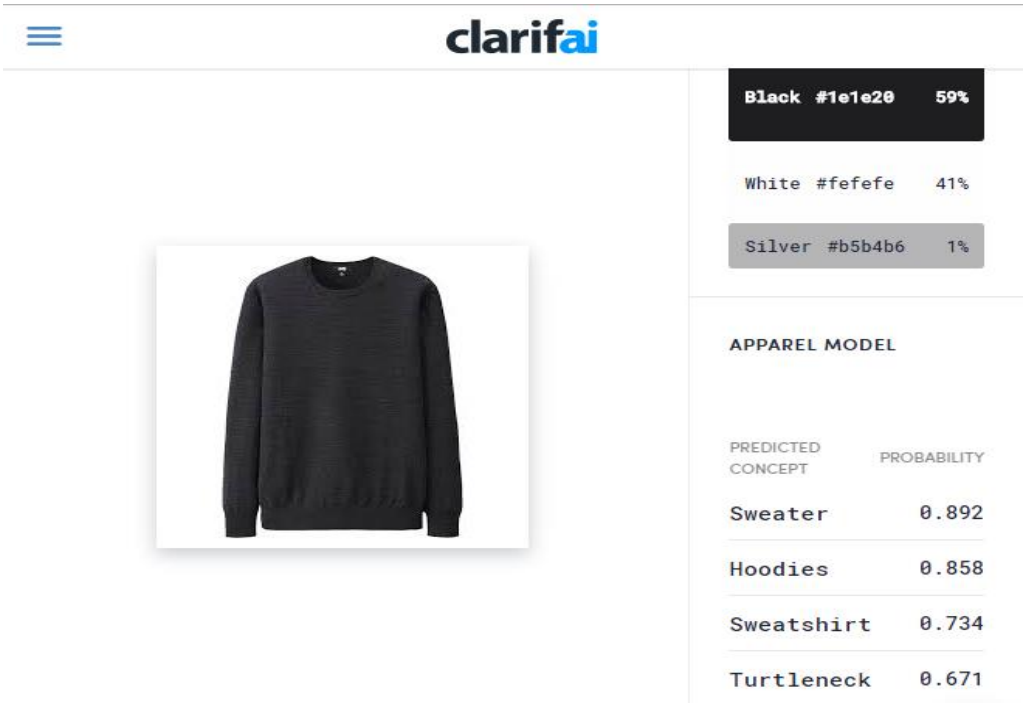
3.1.6 Analisis Data

Analisis data yang terdapat dalam penelitian ini adalah sebagai berikut :

3.1.6.1 Pengambilan Data Clarifai API

Clarifai API memberikan layanan untuk mengenali gambar, dengan memasukkan foto (gambar) sebagai alat *input* dan kemudian Clarifai API akan memberikan daftar prediksi berdasarkan probabilitas tertinggi yang sesuai dengan gambar. Jenis prediksi ditentukan berdasarkan jenis model yang dipilih pada saat akan memasukkan foto, yang mana model tersebut berisi kumpulan konsep yang terkandung dalam foto tersebut. API diakses melalui HTTPS dan diakses melalui *domain* https://api.clarifai.com.

Berikut merupakan contoh pemberian tag prediksi oleh Clarifai berdasarkan foto (gambar) yang menjadi *input* dengan model prediksi *apparel.*



Gambar 3. 3 Contoh Pengenalan Gambar Menggunakan Clarifai

Berdasarkan hasil pemberian tag prediksi oleh Clarifai didapatkan empat tag prediksi beserta nilai probabilitas yang dimiliki dari masing-masing tag prediksi.

1. API Clients

Instruksi untuk proses instalasi *client* API adalah sebagai berikut:

```
# Pip install the client:
  # pip install clarifai


  #The package will be accessible by importing clarifai:

  from clarifai import rest
  from clarifai.rest import ClarifaiApp

  # The client takes the `APP_ID` and `APP_SECRET` you created in your Clarifai
  # account. You can set these variables in your environment as:

  # - `CLARIFAI_APP_ID`
  # - `CLARIFAI_APP_SECRET`
```

Gambar 3. 4 Instalasi Client API

2. Proses Autentikasi

Autentikasi ke API dilakukan menggunakan OAuth2 *clients credentials.* Setiap aplikasi yang dibuat harus memiliki *client* ID yang unik dan rahasia, yang mana *client* ID ini akan digunakan untuk pertukaran *Access Token* yang akan digunakan untuk membuat *Authorized* API. OAuth2 *client credentials* terdiri dari tiga komponen utama yaitu:

a. Client_id, merupakan ID yang akan digunakan pada saat aplikasi akan mengakses API.
b. Client_secret, komponen ini memberikan keamanan pada saat memberi otorisasi dengan API. Ini unik dan dihasilkan satu kali untuk setiap aplikasi di akun Anda.
c. Access_token, komponen ini digunakan untuk memberi otorisasi akses ke API. *Access Token* akan kadaluarsa secara berkala dan harus diperbaharui secara berkelanjutan.

3. Retrieve an Access Token

Semua permintaan API harus memasukkan sebuah *access_token*, untuk mendapatkan sebuah *access token*, *client_id* dan *client_secret* harus dikirimkan sebagasi *Basic Auth* ke *endpoint* token.

```
from clarifai.rest import ClarifaiApp

app = ClarifaiApp("{clientId}", "{clientSecret}")

# NOTE: you probably won't have to handle this.
        The API client automatically refreshes your access token
# before making any network calls if it is expired
app.auth.get_token()
```

Gambar 3. 5 Retrieve an Access Token

JSON akan memberikan respon *access_token* yang telah dimasukkan pada saat meminta API. *Access Token* akan kadaluarsa secara berkala dan harus diperbaharui secara berkala. *Access Token* bisa diperbaharui dengan meminta sebuah *Access Token* yang baru.


```
{
    "status": {
      "code": 10000,
      "description": "Success"
    },
    "access_token": "",
    "expires_in": 176400,
    "scope": "api_access_write api_access api_access_read"
  }
```


Gambar 3. 6 Respon JSON

4. Proses Mengirim Foto

Berikut adalah cara mengirim *input* (foto/gambar) untuk diprediksi dalam bentuk model *general*.

```
from clarifai.rest import Image as ClImage

  model = app.models.get('general-v1.3')
  image = ClImage(url='https://samples.clarifai.com/metro-north.jpg')
  model.predict([image])
```

Gambar 3. 7 Proses Mengirim Foto

5. Proses Menerima Prediksi

Berikut adalah cara menerima prediksi dari foto yang menjadi *input* dalam bentuk model *general*.


```
"status": {
  "code": 10000,
  "description": "Ok"
},
"outputs": [
  {
    "id": "ea68cac87c304b28a8046557062f34a0",
    "status": {
      "code": 10000,
      "description": "Ok"
    },
    "created_at": "2016-11-22T16:50:25Z",
    "model": {
      "name": "general-v1.3",
      "id": "aaa03c23b3724a16a56b629203edc62c",
      "created_at": "2016-03-09T17:11:39Z",
      "app_id": null,
      "output_info": {
        "message": "Show output_info with: GET /models/{model_id}/output_info",
        "type": "concept"
      },
      "model_version": {
        "id": "aa9ca48295b37401f8af92ad1af0d91d",
        "created_at": "2016-07-13T01:19:12Z",
        "status": {
          "code": 21100,
          "description": "Model trained successfully"
        }
      }
    },
    "input": {
      "id": "ea68cac87c304b28a8046557062f34a0",
      "data": {
        "image": {
          "url": "https://samples.clarifai.com/metro-north.jpg"
```


Gambar 3. 8 Respon Pemberian Prediksi

```
      }
    }
  },
  "data": {
    "concepts": [
      {
        "id": "ai_HLmqFqBf",
        "name": "train",
        "app_id": null,
        "value": 0.9989112
      },
      {
        "id": "ai_fvlBqXZR",
        "name": "railway",
        "app_id": null,
        "value": 0.9975532
      },
      {
        "id": "ai_Xxjc3MhT",
        "name": "transportation system",
        "app_id": null,
        "value": 0.9959158
      },
      {
        "id": "ai_6kTjGfF6",
        "name": "station",
        "app_id": null,
        "value": 0.992573
      },
      {
        "id": "ai_RRXLczch",
        "name": "locomotive",
        "app_id": null,
        "value": 0.992556
      },
      {
```


Gambar 3. 9 Respon Pemberian Prediksi (Lanjutan)

```
  "id": "ai_VRmbGVWh",
  "name": "travel",
  "app_id": null,
  "value": 0.98789215
},
{
  "id": "ai_SHNDcmJ3",
  "name": "subway system",
  "app_id": null,
  "value": 0.9816359
},
{
  "id": "ai_jlb9q33b",
  "name": "commuter",
  "app_id": null,
  "value": 0.9712483
},
{
  "id": "ai_461GZ4Gm",
  "name": "railroad track",
  "app_id": null,
  "value": 0.9690325
},
{
  "id": "ai_tr0MBp64",
  "name": "traffic",
  "app_id": null,
  "value": 0.9687052
},
{
  "id": "ai_l4WckcJN",
  "name": "blur",
  "app_id": null,
  "value": 0.9667078
},
{
```


Gambar 3. 10 Respon Pemberian Prediksi (Lanjutan)

```
    "id": "ai_2gkfMDsM",
    "name": "platform",
    "app_id": null,
    "value": 0.9624243
  },
  {
    "id": "ai_CpFBRWzD",
    "name": "urban",
    "app_id": null,
    "value": 0.960752
  },
  {
    "id": "ai_786Zr311",
    "name": "no person",
    "app_id": null,
    "value": 0.95864904
  },
  {
    "id": "ai_6lhccv44",
    "name": "business",
    "app_id": null,
    "value": 0.95720303
  },
  {
    "id": "ai_971KsJkn",
    "name": "track",
    "app_id": null,
    "value": 0.9494642
  },
  {
    "id": "ai_WBQfVV0p",
    "name": "city",
    "app_id": null,
    "value": 0.94089437
  },
  {
```


Gambar 3. 11 Respon Pemberian Prediksi (Lanjutan)

```
        },
        {
          "id": "ai_971KsJkn",
          "name": "track",
          "app_id": null,
          "value": 0.9494642
        },
        {
          "id": "ai_WBQfVV0p",
          "name": "city",
          "app_id": null,
          "value": 0.94089437
        },
        {
          "id": "ai_dSCKh8xv",
          "name": "fast",
          "app_id": null,
          "value": 0.9399334
        },
        {
          "id": "ai_TZ3C79C6",
          "name": "road",
          "app_id": null,
          "value": 0.93121606
        },
        {
          "id": "ai_VSVscs9k",
          "name": "terminal",
          "app_id": null,
          "value": 0.9230834
        }
      ]
    }
  }
 ]
}
```


Gambar 3. 12 Respon Pemberian Prediksi (Lanjutan)

3.1.6.2 Pengambilan Data Open Weather API

Open Weather API memberikan layanan untuk mendapatkan informasi cuaca. Proses permintaan informasi cuaca menggunakan *Open Weather* API adalah sebagai berikut.

```
// get the Temperature
weather.getTemperature(function(err, temp){
    console.log(temp);
});

// get the Atm Pressure
weather.getPressure(function(err, pres){
    console.log(pres);
});

// get the Humidity
weather.getHumidity(function(err, hum){
    console.log(hum);
});

// get the Description of the weather condition
weather.getDescription(function(err, desc){
    console.log(desc);
});

// get all the JSON file returned from server (rich of info)
weather.getAllWeather(function(err, JSONObj){
    console.log(JSONObj);
});
```

Gambar 3. 13 Function untuk Meminta Data

Fungsi ini akan memberikan balasan berupa sebuah JSON *object openweathermap.org* yang lengkap. JSON *Object* ini mudah untuk digunakan dan mampu memberikan setiap kemungkinan yang bisa digunakan untuk memberikan data cuaca.

```
{"coord":{"lon":139,"lat":35},
"sys":{"country":"JP","sunrise":1369769524,"sunset":1369821049},
"weather":[{"id":804,"main":"clouds","description":"overcast clo
"main":{"temp":289.5,"humidity":89,"pressure":1013,"temp_min":28
"wind":{"speed":7.31,"deg":187.002},
"rain":{"3h":0},
"clouds":{"all":92},
"dt":1369824698,
"id":1851632,
"name":"Shuzenji",
"cod":200}
```


Gambar 3. 14 JSON Object Open Weather API

3.1.6.3 Pengambilan Data Instagram API

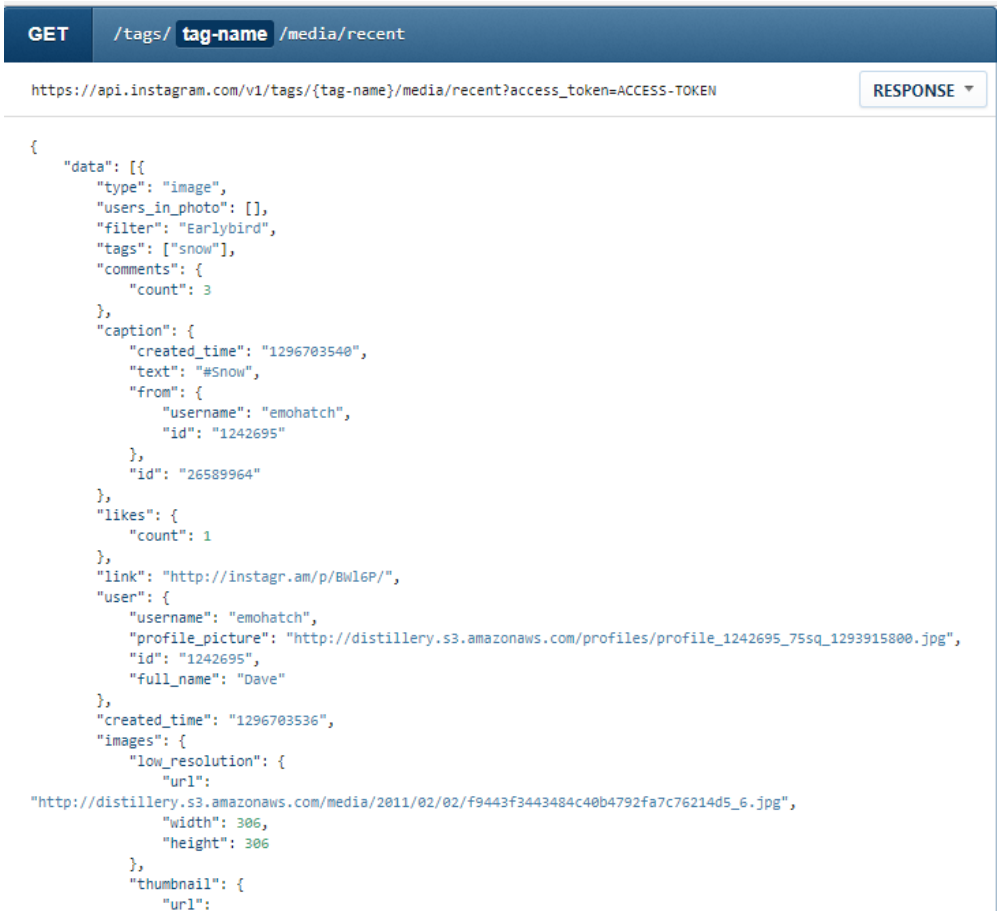
Instagram API hanya bisa diakses melalui *https* dan *api.instagram.com*. Instagram API memerlukan sebuah *access_token*_dari pengguna yang telah diotentikasi pada setiap *endpoint.*

1. Search For Tags by Name



Gambar 3. 15 Response Get Tags by Name

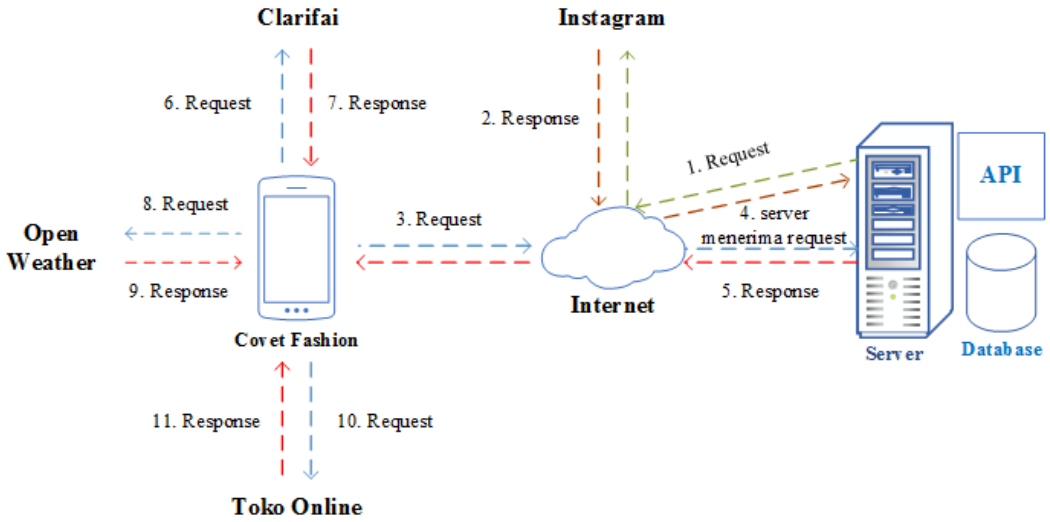
2. Get a List of Recently Tagged Media



Gambar 3. 16 Response Get Tags Recent Media

3.1.7 Analisis Arsitektur Sistem

Platform mobile adalah sistem yang dipilih untuk pembangunan dari perangkat lunak ini. Arsitektur perangkat lunak pada *platform mobile* menggambarkan bagaimana perangkat lunak saling berinteraksi seperti diilustrasikan pada gambar berikut:



Gambar 3. 17 Arsitektur Perangkat Lunak Pada Platform Mobile

Berikut adalah deskripsi dari Gambar 3.17 Arsitektur Perangkat Lunak Pada *Platform Mobile*:

1. *Server* melakukan *request* ke Instagram
2. Instagram memberikan *response* kepada *server* melalui API.
3. Perangkat *mobile* pengguna melakukan *request* data ke *server*.
4. *Server* menerima *request* data dari perangkat *mobile*.
5. *Server* memberikan *response* dengan memberikan data dalam bentuk JSON ke perangkat *mobile*.
6. Perangkat *mobile* melakukan *request* ke Clarifai melalui API.
7. Clarifai memberikan *response* kepada *server* melalui API.
8. Perangkat *mobile* melakukan *request* ke Open Weather melalui API.
9. Open Weather memberikan *response* kepada *server* melalui API.
10. Perangkat *mobile* melakukan *request* ke Toko *online* melalui API.

11. Toko *online* memberikan *response* kepada *server* melalui API.

3.1.8 Spesifikasi Kebutuhan Perangkat lunak

Spesifikasi kebutuhan perangkat lunak terbagi kedalam dua kebutuhan yaitu kebutuhan non fungsional dan fungsional. Kebutuhan fungsional dapat dilihat pada tabel di bawah ini.

Tabel 3. 19 Spesifikasi Kebutuhan Fungsional

Kode SKPL	Spesifikasi Kebutuhan Perangkat Lunak
SKPL-F-001	Sistem *mobile* menyediakan fasilitas bagi *user* untuk melihat rekomendasi OOTD berdasarkan *event*
SKPL-F-002	Sistem *mobile* menyediakan fasilitas bagi *user* untuk melihat rekomendasi OOTD berdasarkan *mix and match* dari foto *item fashion*
SKPL-F-003	Sistem *mobile* menyediakan fasilitas bagi *user* untuk melihat foto dengan #OOTD terbaru dari Instagram
SKPL-F-004	Sistem *mobile* menyediakan fasilitas bagi *user* untuk melihat informasi cuaca pada saat itu
SKPL-F-005	Sistem *mobile* menyediakan fasilitas bagi *user* untuk melihat hasil tag
SKPL-F-006	Sistem *mobile* menyediakan fasilitas bagi *user* untuk melihat detail *item*
SKPL-F-007	Sistem *mobile* menyediakan fasilitas memanggil URL Open Weather API
SKPL-F-008	Sistem *mobile* menyediakan fasilitas memanggil URL Toko *Online* API
SKPL-F-009	Sistem *mobile* menyediakan fasilitas memanggil URL data Clarifai API
SKPL-F-010	Sistem *mobile* menyediakan fasilitas memanggil URL data Facebook API

Kebutuhan non fungsional pada perangkat lunak yang dibangun dapat dilihat pada tabel di bawah ini.

Tabel 3. 20 Spesifikasi Kebutuhan Non Fungsional

Kode SKPL	Spesifikasi Kebutuhan Perangkat Lunak
SKPL-NF-001	Sistem yang dibangun yaitu berbasis *mobile*
SKPL-NF-002	Sistem yang dibangun minimal menggunakan sistem operasi android versi 5.0 Lollipop
SKPL-NF-003	Sistem dibangun dengan spesifikasi *hardware* yang memenuhi standar minimum kebutuhan.
SKPL-NF-004	Perangkat lunak yang dibangun dapat menampilkan rekomendasi OOTD berdasarkan foto dan data *event*
SKPL-NF-005	Desain *user interface* pada perangkat lunak yang akan dibangun mengacu kepada *design guidelines* android dari google
SKPL-NF-006	Perangkat lunak yang dibangun harus menghasilkan format *file* standard (.json) yang bisa digunakan oleh pihak luar yang berkepentingan.

3.1.9 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional pada sistem ini meliputi analisis perangkat lunak, analisis perangkat keras dan perangkat pikir.

3.1.9.1 Produk Requirement

Berikut adalah kebutuhan non fungsional berdasarkan klasifikasi *product requirement.*

1. SKPL-NF-001 Sistem yang dibangun yaitu berbasis *mobile.*

2. SKPL-NF-002 Sistem yang dibangun minimal menggunakan sistem operasi android versi 4.1 *Jelly Bean*, dan dibangun dengan kebutuhan perangkat lunak minimum.

Tabel 3. 21 SKPL-NF-002

Sistem Operasi	Platform Bebas
Web server	XAMPP *version* 1.1.7
Kode editor	*Sublime text*
IDE (*Integrated Development Environment*)	Android Studio
ADT (*Android Development Tools*)	*Version* 23.1.6
Android Software Development Kit	*Android Studio Framework*

3. SKPL-NF-003 Sistem dibangun dengan spesifikasi *hardware* yang memenuhi standar minimum kebutuhan seperti pada tabel berikut.

Tabel 3. 22 SKPL-NF-003

Jenis	Spesifikasi
VGA Komputer	512 MB
Resolusi layar	1024 x 600
RAM komputer	2GB
RAM Android	512 MB
Resolusi Android	320 x 480

4. SKPL-NF-004 Perangkat lunak yang dibangun dapat menampilkan rekomendasi OOTD berdasarkan foto dan data *event*

3.1.9.2 Organisational Requirement

Kebutuhan non fungsional berdasarkan klasifikasi *Organisational Requirement* adalah SKPL-NF-005, desain *user interface* pada perangkat lunak yang akan dibangun mengacu kepada *design guidelines* android dari google.

3.1.9.3 External Requirement

Kebutuhan non fungsional berdasarkan klasifikasi *external requirement* adalah SKPL-NF-006, perangkat lunak yang dibangun harus menghasilkan format *file* standar (.json) yang bisa digunakan oleh pihak luar yang berkepentingan.

3.1.9.4 Analisis Kebutuhan Perangkat Pikir

Analisis kebutuhan perangkat piker adalah batasan-batasan dari layanan dan fungsi-fungsi yang dibutuhkan oleh sistem dari segi perangkat piker. Pada penelitian ini analisis kebutuhan perangkat piker dikelompokkan menjadi tiga kategori yaitu *knowledge and experience, job and task characteristic* dan *physical characteristic.*

1. *Knowledge and Experience*

Knowledge and experience yang diharapkand ari pengguna perangkat lunak pada penelitian ini adalah sebagai berikut.

Tabel 3. 23 Knowledge and Experience

Knowledge and Experience	Level
Computer literacy	Cukup sampai tinggi
system experience	Rendah sampai tinggi
Application experience	Rendah sampai tinggi
Task experience	Rendah sampai tinggi
Other system use	*Frequent*
Education	Perguruan tinggi keatas
Reading level	Sedang sampai tinggi
Typing skill	Sedang sampai tinggi
Native language or culture	*English* / Indonesia

2. *Job and Task Characteristic*

Job and characteristic yang diharapkan dari pengguna perangkat lunak pada penelitian ini adalah sebagi berikut.

Tabel 3. 24 Job and Task Characteristic

Job and Task Characteristic	Level
Task structure	Tinggi
Social interactions	Perlu
Primary training	*None*
Jab category	*General*
Frequency of use	Rendah
Task or need importance	Sedang

3. *Physical Characteristic*

Physical characteristic yang diharapkan dari pengguna perangkat lunak pada penelitian ini adalah sebagai berikut.

Tabel 3. 25 Physical Characteristic

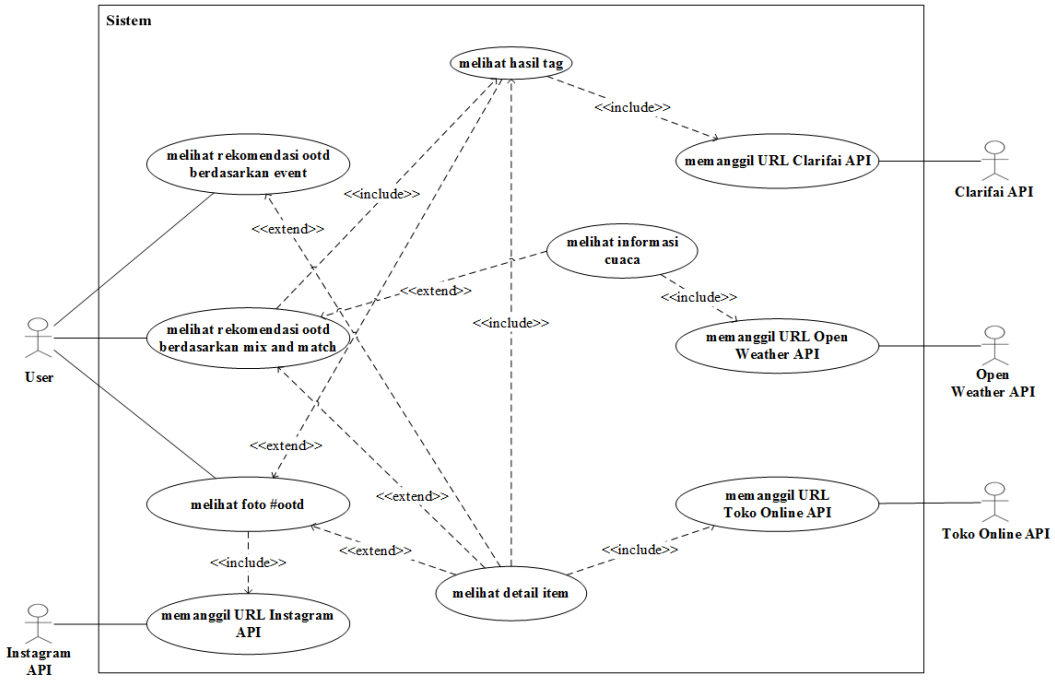
Physical Characteristic	
Age	*Middle aged and elderly*
Gender	*Male and female*
Handedness	*Both*
Disabilities	*None*

3.1.10 Analisis Kebutuhan Fungsional

Analisis sistem dimodelkan dengan menggunakan UML (*Unified Modelling Language*) meliputi *use case diagram*, *use case scenario*, *activity diagram*, *sequence diagram*, dan *class diagram*. Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sistem dan menjelaskan kebutuhan yang diperlukan agar sistem dapat berjalan dengan baik serta sesuai dengan kebutuhan.

3.1.10.1 Use Case Diagram

Use case diagram adalah diagram yang menunjukkan fungsionalitas suatu sistem atau kelas, bagaimana sistem tersebut berinteraksi dan menjelaskan sistem secara fungsional yang terlihat oleh pengguna. Berikut ini adalah diagram *use case* dari aplikasi rekomendasi OOTD (*Outfit of The Day*) berdasarkan *event* dan *mix and match item*.



Gambar 3. 18 Use Case Diagram

Definisi aktor dan deskripsi *role* untuk aktor di dalam sistem dapat dilihat table sebagai berikut :

Tabel 3. 26 Definisi Aktor

No.	Aktor	Deskripsi
1	*User*	Aktor dengan *role* ini memiliki wewenang untuk menggunakan fungsionalitasnya pada level *user interface*
2	Open weather API	Aktor dengan *role* ini memiliki wewenang sebagai penyedia API atau jembatan sumber data *Open Weather*
3	Clarifai API	Aktor dengan *role* ini memiliki wewenang sebagai penyedia API atau jembatan sumber data Clarifai
4	Instagram API	Aktor dengan *role* ini memiliki wewenang sebagai penyedia API atau jembatan sumber data Instagram
5	Toko *online* API	Aktor dengan *role* ini memiliki wewenang sebagai penyedia API atau jembatan sumber data toko *online*

Identifikasi *use case* yang terdapat dalam sistem dapat dilihat pada tabel sebagai berikut :

Tabel 3. 27 Identifikasi Use Case

No.	*Use Case*	Deskripsi
UC-01	Melihat rekomendasi OOTD berdasarkan *event*	Sistem menampilkan rekomendasi OOTD berdasarkan *event*
UC-02	Melihat rekomendasi OOTD berdasarkan *mix and match*	Sistem menampilkan rekomendasi OOTD berdasarkan *mix and match item*

	item	
UC-03	Melihat foto #OOTD	Sistem meanmpilkan foto dengan #OOTD terbaru dari Instagram
UC-04	Melihat hasil *tag*	Sistem menampilkan hasil tag
UC-05	Melihat informasi cuaca	Sistem menampilkan informasi cuaca pada saat itu
UC-06	Melihat detail *item*	Sitem menampilkan detail dari *item* yang direkomendasikan
UC-07	Memanggil URL Instagram API	Sistem memanggil URL Instagram API
UC-08	Memanggil URL Clarifai API	Sistem memanggil URL Clarifai API
UC-09	Memanggil URL Open Weather API	Sistem memanggil URL Open Weather API
UC-10	Memanggil URL Toko *online* API	Sistem memanggil URL Toko *online* API

3.1.10.2 Use Case Scenario

Use case scenario bertujuan untuk mendeskripsikan urutan langkah-langkah dalam proses yang dilakukan oleh aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor. Berdasarkan *use case diagram* pada **Gambar 3.17 Use Case Diagram,** maka *use case scenario* dapat dijelaskan sebagai berikut.

1. Use Case Scenario Melihat Rekomendasi OOTD Berdasarkan Event

Use case scenario melihat rekomendasi OOTD berdasarkan *event* dapat dilihat pada tabel sebagai berikut:

Tabel 3. 28 Use Case Scenario Melihat Rekomendasi OOTD Berdasarkan Event

Use Case Name	**Melihat Rekomendasi OOTD Berdasarkan Event**	
Related Requirements	-	
Goal In Context	Melihat rekomendasi OOTD berdasarkan *event*	
Preconditions	Halaman utama ditampilkan	
Successful End Condition	Rekomendasi OOTD berdasarkan *event* berhasil ditampilkan	
Failed End Condition	Rekomendasi OOTD berdasarkan *event* tidak berhasil ditampilkan	
Primary Actors	*User*	
Trigger	*User* memilih menu rekomendasi berdasarkan *event*	
Main Flow	**Step**	**Action**
	1	Sistem menampilkan halaman *form* rekomendasi berdasarkan *event*
	2	*User* mengisi form cari rekomendasi untuk *event*
	3	Sistem melakukan pencocokan data di dalam *database*

	4	Sistem menmapilkan rekomendasi OOTD berdasarkan *event*
Extensions	**Step**	**Branching Action**
	4.1	Sistem gagal menampilkan rekomendasi OOTD berdasarkan *event*
	4.2	Sistem berhasil menampilkan rekomendasi OOTD berdasarkan *event*

2. Use Case Scenario Melihat Rekomendasi OOTD Berdasarkan Mix and Match Item

Use case scenario melihat rekomendasi OOTD berdasarkan *mix and match item* dapat dilihat pada tabel sebagai berikut:

Tabel 3. 29 Use Case Scenario Melihat Rekomendasi OOTD Berdasarkan Mix and Match Item

Use Case Name	**Melihat Rekomendasi OOTD Berdasarkan Mix and Match Item**	
Related Requirements	-	
Goal In Context	Melihat rekomendasi OOTD berdasarkan *mix and match item*	
Preconditions	Halaman utama ditampilkan	
Successful End Condition	Rekomendasi OOTD berdasarkan *mix and match item* berhasil ditampilkan	
Failed End Condition	Rekomendasi OOTD berdasarkan *mix and match item* tidak berhasil ditampilkan	
Primary Actors	*User*	
Trigger	*User* memilih menu rekomendasi berdasarkan *mix and match item*	
Main Flow	**Step**	**Action**
	1	Sistem menampilkan fitur kamera dari *device*
	2	*User* mengambil gambar
	3	Clarifai mengolah gambar
	4	*User* memilih *tag* prediksi hasil *scan* gambar
	5	Sistem mengecek rekomendasi OOTD di dalam *database*
	6	Sistem menampilkan rekomendasi OOTD berdasarkan *mix and match item*
Extensions	**Step**	**Branching Action**
	6.1	Sistem gagal menampilkan rekomendasi OOTD berdasarkan *mix and match item*
	6.2	Sistem berhasil menampilkan rekomendasi OOTD berdasarkan *mix and match item*

3. Use Case Scenario Melihat Foto #OOTD

Use case scenario melihat foto #OOTD dapat dilihat pada tabel sebagai berikut:

Tabel 3. 30 Use Case Scenario Melihat Foto #OOTD

Use Case Name	**Melihat Foto #OOTD**	
Related Requirements	-	
Goal In Context	Melihat foto dengan #OOTD terbaru dari Instagram	
Preconditions	Halaman utama ditampilkan	
Successful End Condition	Foto #OOTD berhasil ditampilkan	
Failed End Condition	Foto #OOTD tidak berhasil ditampilkan	
Primary Actors	Sistem	
Trigger	-	
Main Flow	**Step**	**Action**
	1	Sistem menampilkan foto #OOTD
Extensions	**Step**	**Branching Action**
	1.1	Sistem gagal menampilkan foto #OOTD
	1.2	Sistem berhasil menampilkan foto #OOTD

4. Use Case Scenario Melihat Hasil Tag

Use case scenario melihat hasil *tag* dapat dilihat pada tabel sebagai berikut:

Tabel 3. 31 Use Case Scenario Melihat Hasil Tag

Use Case Name	**Melihat Hasil Tag**	
Related Requirements		
Goal In Context	*User* melihat hasil tag	
Preconditions	Hasil *tag ditampilkan*	
Successful End Condition	Hasil *tag* berhasil ditampilkan	
Failed End Condition	Hasil *tag* tidak berhasil ditampilkan	
Primary Actors	*User*	
Trigger	*User* melihat hasil *tag*	
Main Flow	**Step**	**Action**
	1	Sistem menampilkan halaman hasil *tag*
	2	*User* melihat hasil *tag*
Extensions	**Step**	**Branching Action**
	1.1	Sistem gagal menampilkan hasil *tag*
	1.2	Sistem berhasil menampilkan hasil *tag*

5. Use Case Scenario Melihat Informasi Cuaca

Use case scenario melihat informasi cuaca dapat dilihat pada tabel sebagai berikut:

Tabel 3. 32 Use Case Scenario Melihat Informasi Cuaca

Use Case Name	**Melihat Informasi Cuaca**	
Related Requirements	-	
Goal In Context	*User* melihat informasi cuaca	
Preconditions	Rekomendasi OOTD berdasarkan *mix and match item* telah ditampilkan	
Successful End Condition	Informasi cuaca pada saat itu berhasil ditampilkan	
Failed End Condition	Informasi cuaca pada saat itu tidak berhasil ditampilkan	
Primary Actors	*User*	
Trigger	*User* memilih menu informasi cuaca	
Main Flow	**Step**	**Action**
	1	Halaman rekomendasi OOTD berdasarkan *mix and match* telah berhasil ditampilkan
	2	*User* memilih menu informasi cuaca
	3	Sistem menampilkan informasi cuaca
Extensions	**Step**	**Branching Action**
	3.1	Sistem gagal menampilkan informasi cuaca
	3.2	Sistem berhasil menampilkan informasi cuaca

6. Use Case Scenario Melihat Detail Item

Use case scenario melihat detail *item* dapat dilihat pada tabel sebagai berikut:

Tabel 3. 33 Use Case Scenario Detail Item

Use Case Name	**Melihat Detail Item**	
Related Requirements	-	
Goal In Context	*User* melihat detail *item*	
Preconditions	Rekomendasi OOTD telah ditampilkan	
Successful End Condition	Sistem menampilkan informasi detail *item* rekomendasi	
Failed End Condition	Sistem gagal menampilkan informasi detail *item* rekomendasi	
Primary Actors	*User*	
Trigger	*User* memilih item	
Main Flow	**Step**	**Action**
	1	*User* memilih rekomendasi
	2	Sistem menampilkan detail *item* rekomendasi

Extensions	**Step**	**Branching Action**
	2.1	Sistem gagal menampilkan detail *item* rekomendasi
	2.2	Sistem berhasil menampilkan detail *item* rekomendasi

7. Use Case Scenario Memanggil URL Instagram

Use case scenario memanggil URL Instagram dapat dilihat pada tabel sebagai berikut:

Tabel 3. 34 Use Case Scenario Memanggil URL Instagram

Use Case Name	**Memanggil URL Instagram**	
Related Requirements	-	
Goal In Context	Sistem memanggil URL Instagram	
Preconditions	Sistem memanggil URL Instagram	
Successful End Condition	URL Instagram berhasil dipanggil	
Failed End Condition	URL Instagram gagal dipanggil	
Primary Actors	Sistem	
Trigger	Sistem memanggil URL Instagram	
Main Flow	**Step**	**Action**
	1	Sistem memanggil URL Instagram
	2	Instagram memberikan *response*
Extensions	**Step**	**Branching Action**
	1.1	Sistem gagal memanggil URL Instagram
	1.2	Sistem berhasil memanggil URL Instagram

8. Use Case Scenario Memanggil URL Clarifai

Use case scenario memanggil URL Clarifai dapat dilihat pada tabel sebagai berikut:

Tabel 3. 35 Use Case Scenario Memanggil URL Clarifai

Use Case Name	**Mengambil Data Cuaca Melalui API**	
Related Requirements	-	
Goal In Context	Sistem memanggil URL Clarifai	
Preconditions	Sistem memanggil URL Clarifai	
Successful End Condition	URL Clarifai berhasil dipanggil	
Failed End Condition	URL Clarifai gagal dipanggil	
Primary Actors	Sistem	
Trigger	-	
Main Flow	**Step**	**Action**
	1	Sistem memanggil URL Clarifai
	2	Clarifai memberikan memberikan objek JSON

Extensions	**Step**	**Branching Action**
	1.1	Sistem gagal memanggil URL Clarifai
	1.2	Sistem berhasil memanggil URL Clarifai

9. Use Case Scenario Memanggil URL Open Weather

Use case scenario memanggil URL Open Weather dapat dilihat pada tabel sebagai berikut:

Tabel 3. 36 Use Case Scenario Memanggil URL Open Weather

Use Case Name	**Memanggil URL Open Weather**	
Related Requirements	-	
Goal In Context	Sistem memanggil URL Open Weather	
Preconditions	Sistem memanggil URL Open Weather	
Successful End Condition	URL Open Weather berhasil dipanggil	
Failed End Condition	URL Open Weather gagal dipanggil	
Primary Actors	Sistem	
Trigger	Sistem memanggil URL Open Weather	
Main Flow	**Step**	**Action**
	1	Sistem memanggil URL Open Weather
	2	Open Weather memberikan objek JSON
Extensions	**Step**	**Branching Action**
	1.1	Sistem gagal memanggil URL Open Weather
	1.2	Sistem berhasil memanggil URL Open Weather

10. Use Case Scenario Memanggil URL Toko Online

Use case scenario mengambil URL toko *online* dapat dilihat pada tabel sebagai berikut:

Tabel 3. 37 Use Case Scenario Memanggil URL Toko Online

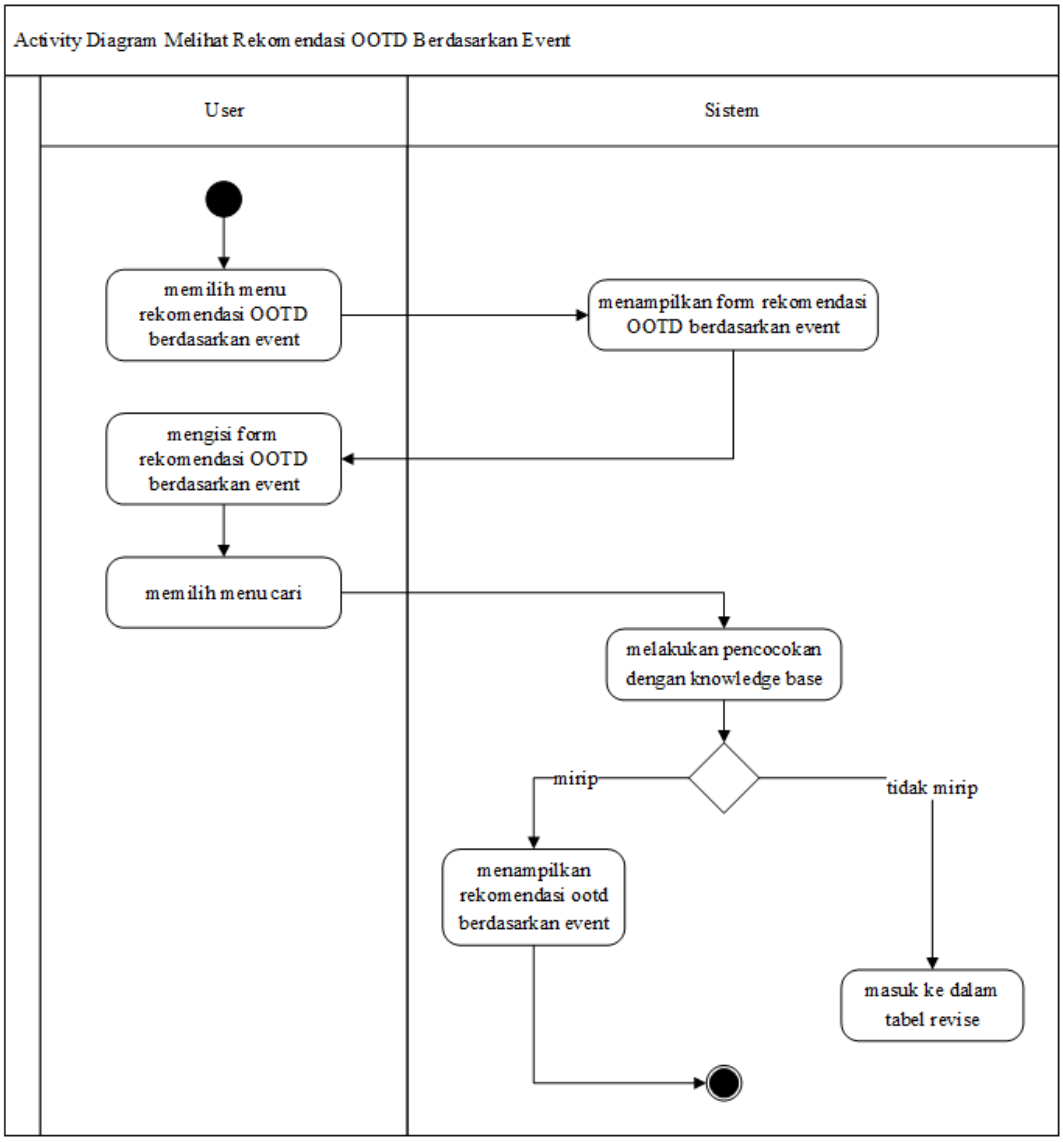
Use Case Name	**Memanggil URL Toko Online**	
Related Requirements	-	
Goal In Context	Sistem memanggil URL Toko *online*	
Preconditions	Sistem memanggil URL Toko *online*	
Successful End Condition	URL Toko *online* berhasil dipanggil	
Failed End Condition	URL Toko *online* gagal dipanggil	
Primary Actors	Sistem	
Trigger	-	
Main Flow	**Step**	**Action**
	1	Sistem memanggil URL toko *online*
	2	Toko *online* memberikan objek JSON
Extensions	**Step**	**Branching Action**
	1.1	Sistem gagal memanggil URL toko *online*
	1.2	Sistem berhasil memanggil URL toko *online*

3.1.10.3 Activity Diagram

Activity diagram merupakan diagram yang memodelkan aliran kerja atau workflow dari urutan aktifitas dalam suatu proses yang mengacu pada use case diagram yang ada. Berikut ini adalah penjelasan dari masing-masing *activity diagram*.

1. Activity Diagram Melihat Rekomendasi OOTD Berdasarkan Event

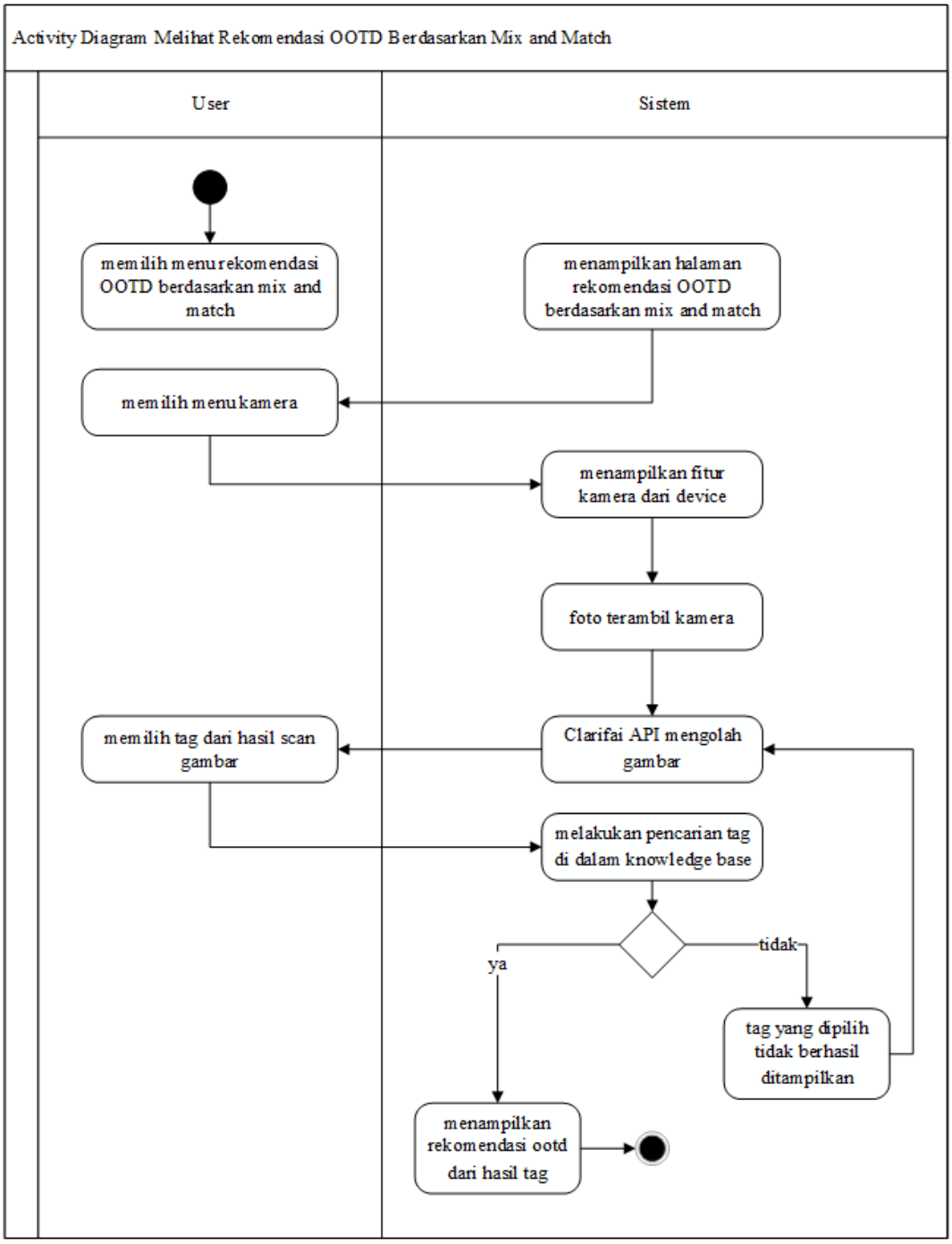
Activity diagram melihat rekomendasi OOTD berdasarkan *event* dapat dilihat pada gambar sebagai berikut:



Gambar 3. 19 Activity Diagram Melihat Rekomendasi OOTD Berdasarkan Event

2. Activity Diagram Melihat Rekomendasi OOTD Berdasarkan Mix and Match Item

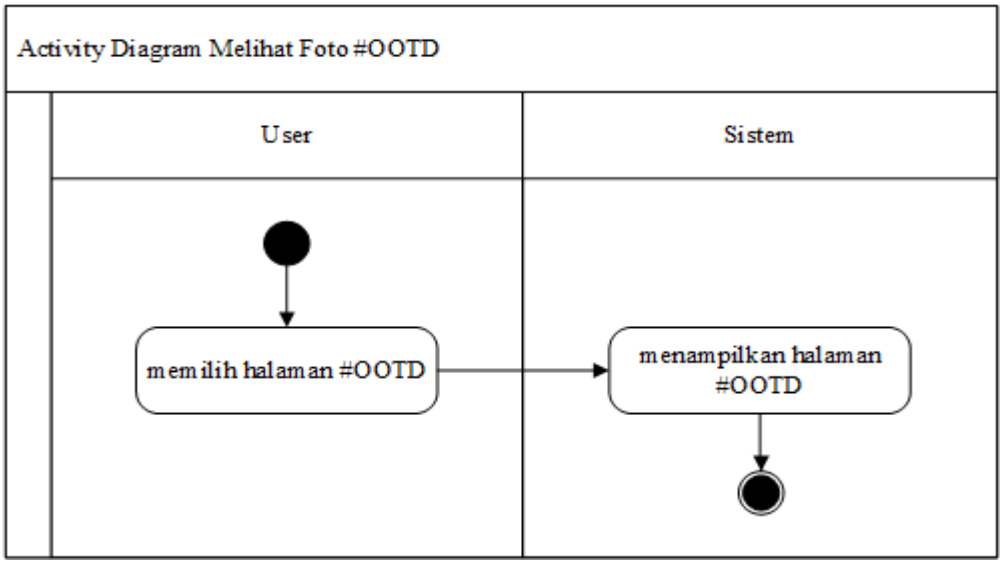
Activity diagram melihat rekomendasi ootd berdasarkan *mix and match* item dapat dilihat pada gambar sebagai berikut:



Gambar 3. 20 Activity Diagram Melihat Rekomendasi OOTD Berdasarkan Mix and Match Item

3. Activity Diagram Melihat Foto #OOTD

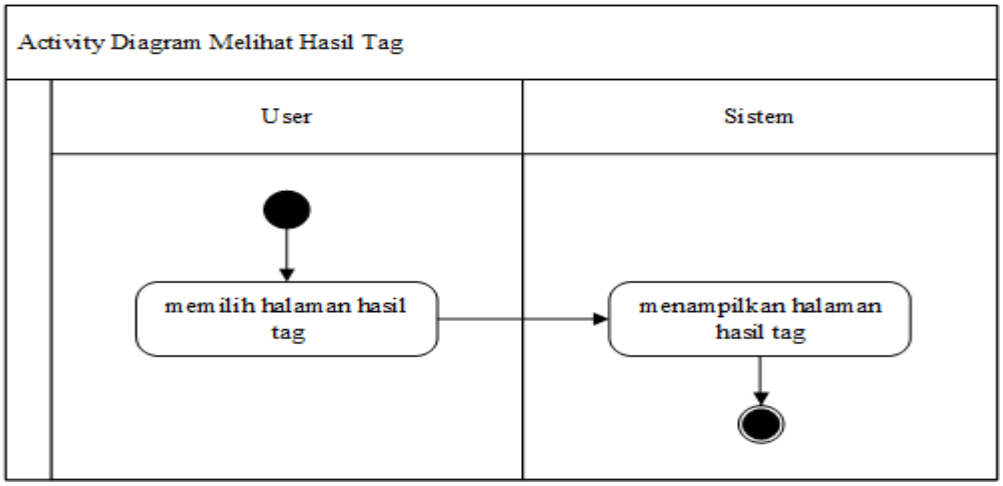
Activity diagram melihat foto #OOTD dapat dilihat pada gambar sebagai berikut:



Gambar 3. 21 Activity Diagram Melihat Foto #OOTD

4. Activity Diagram Melihat Hasil Tag

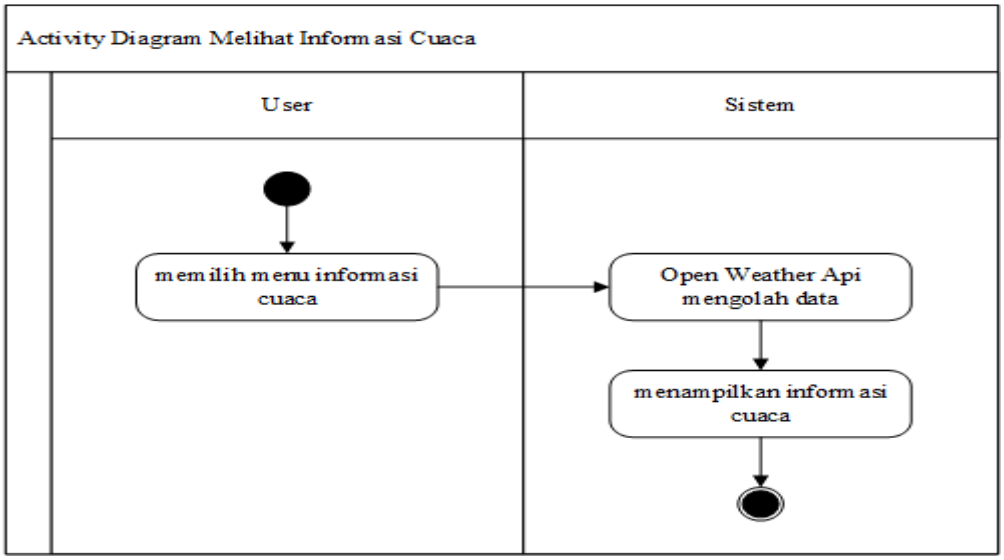
Activity diagram melihat hasil tag dapat dilihat pada gambar sebagai berikut:



Gambar 3. 22 Activity Diagram Melihat Hasil Tag

5. Activity Diagram Melihat Informasi Cuaca

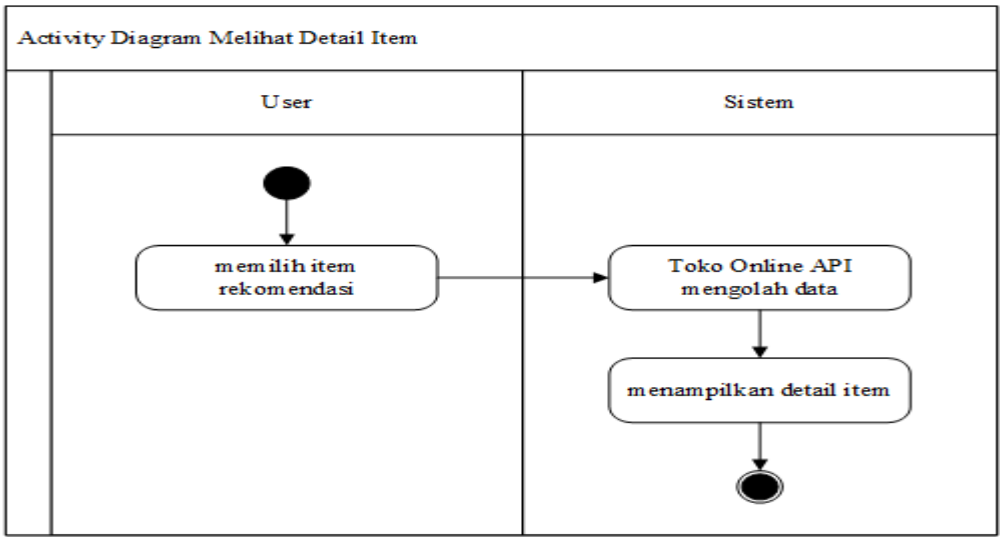
Activity diagram melihat informasi cuaca dapat dilihat pada gambar sebagai berikut:



Gambar 3. 23 Activity Diagram Melihat Informasi Cuaca

6. Activity Diagram Melihat Detail Item

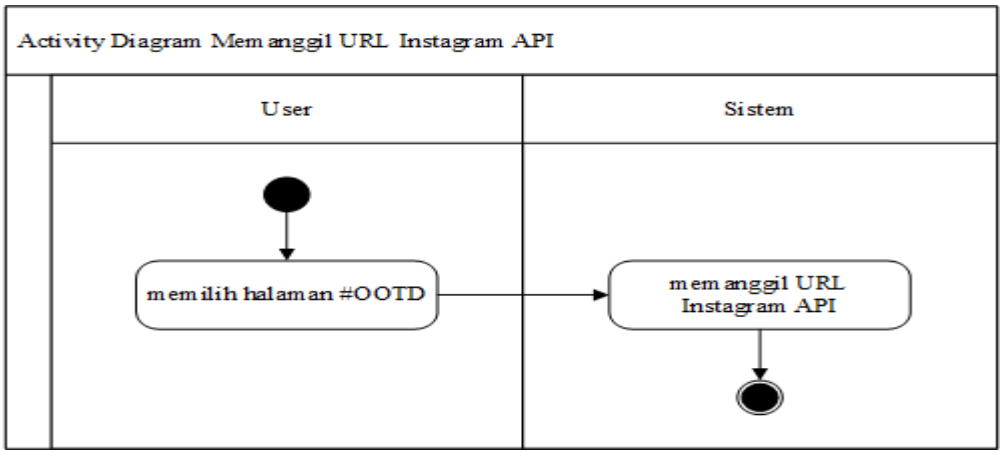
Activity diagram melihat detail item dapat dilihat pada gambar sebagai berikut:



Gambar 3. 24 Activity Diagram Melihat Detail Item

7. Activity Diagram Memanggil URL Instagram

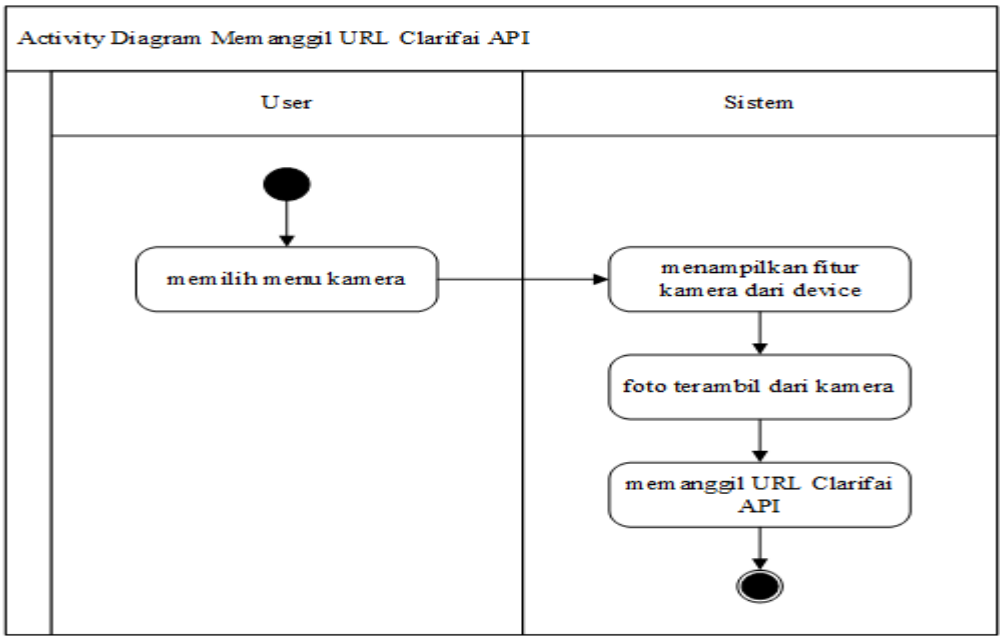
Activity diagram memanggil URL Instagram dapat dilihat pada gambar sebagai berikut:



Gambar 3. 25 Activity Diagram Memanggil URL Instagram

8. Activity Diagram Memanggil URL Clarifai

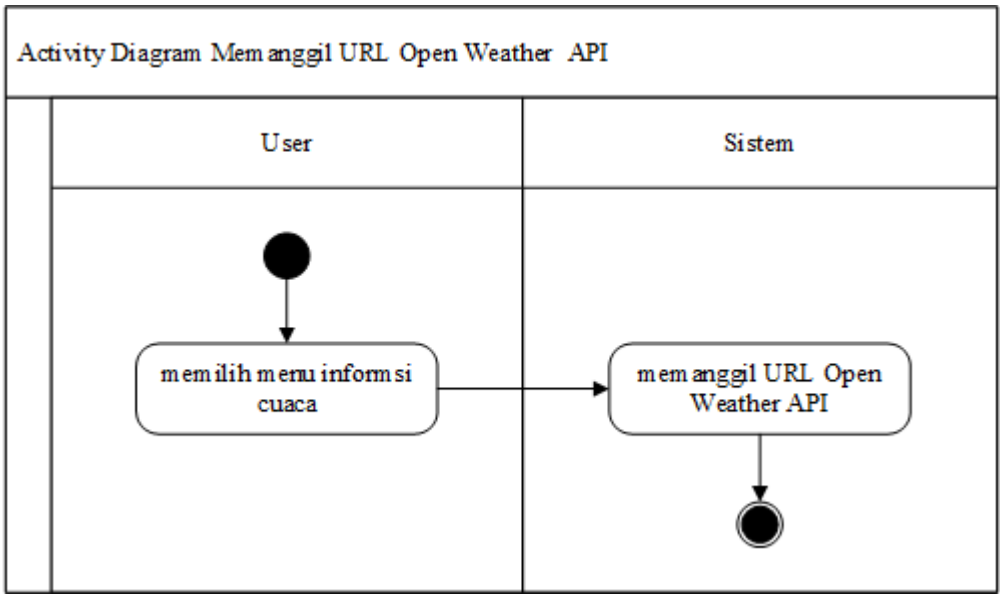
Activity diagram memanggil URL Clarifai dapat dilihat pada gambar sebagai berikut:



Gambar 3. 26 Activity Diagram Memanggil URL Clarifai

9. Activity Diagram Memanggil URL Open Weather

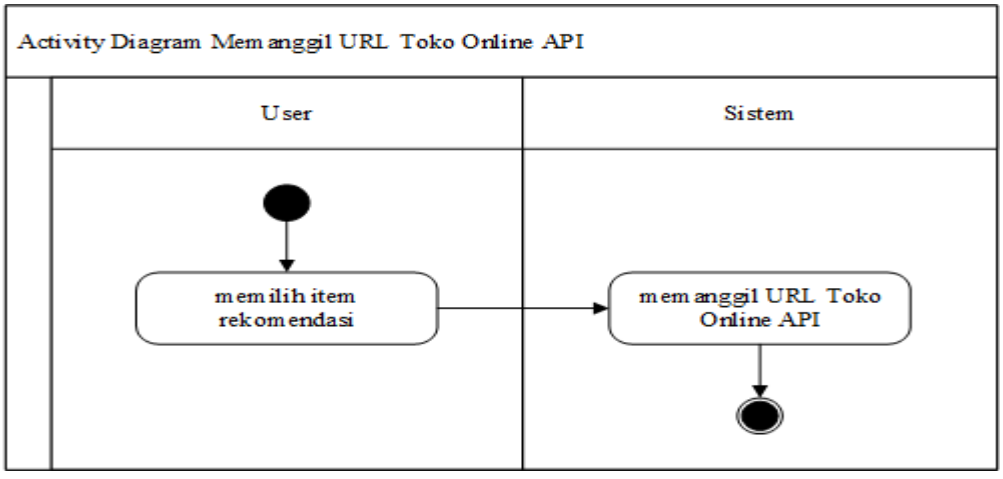
Activity diagram memanggil URL Open Weather dapat dilihat pada gambar sebagai berikut:



Gambar 3. 27 Activity Diagram Memanggil URL Clarifai

10. Activity Diagram Memanggil URL Toko Online

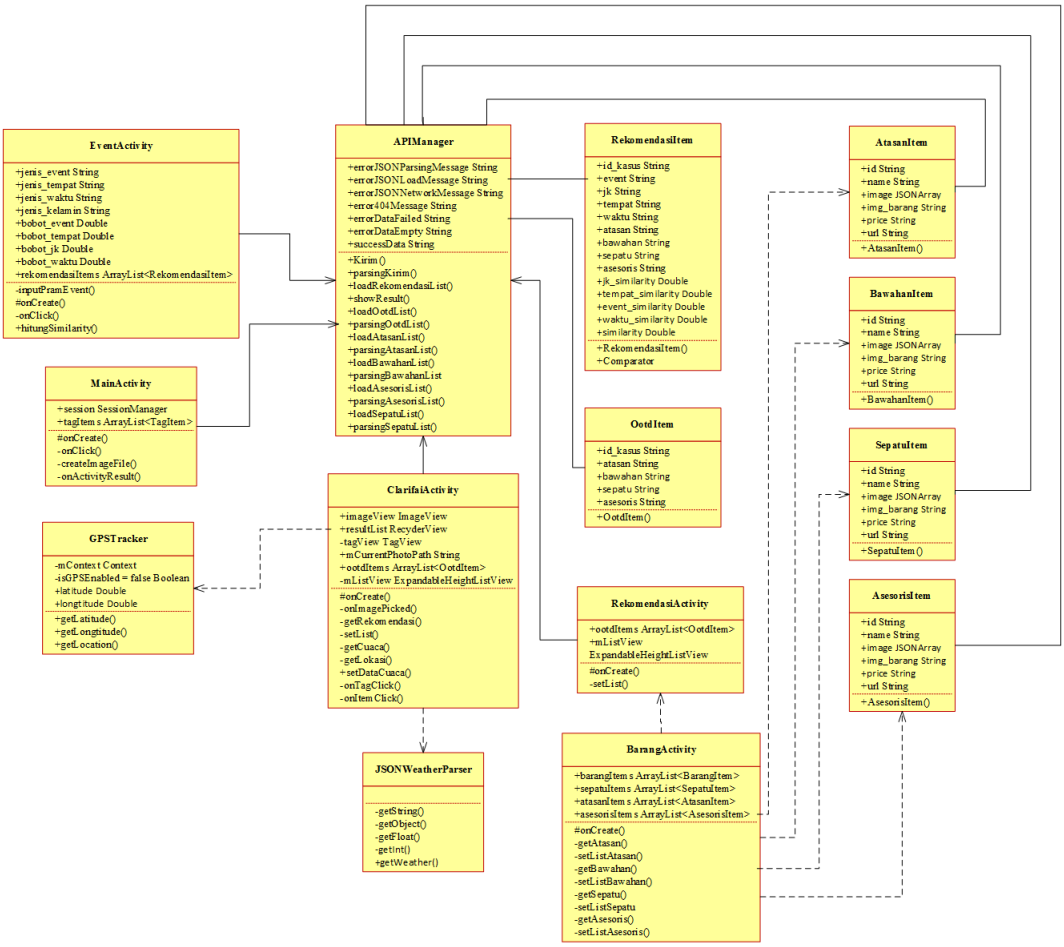
Activity diagram memanggil URL toko *online* dapat dilihat pada gambar sebagai berikut:



Gambar 3. 28 Activity Diagram Memanggil URL Toko Online

3.1.10.4 Class Diagram

Class Diagram menggambarkan keadaan suatu sistem (atribut), dan memberikan pelayanan untuk menyelesaikan keadaan tersebut (metoda). Berikut adalah gambar dari *class diagram* pada sistem yang akan dibangun :



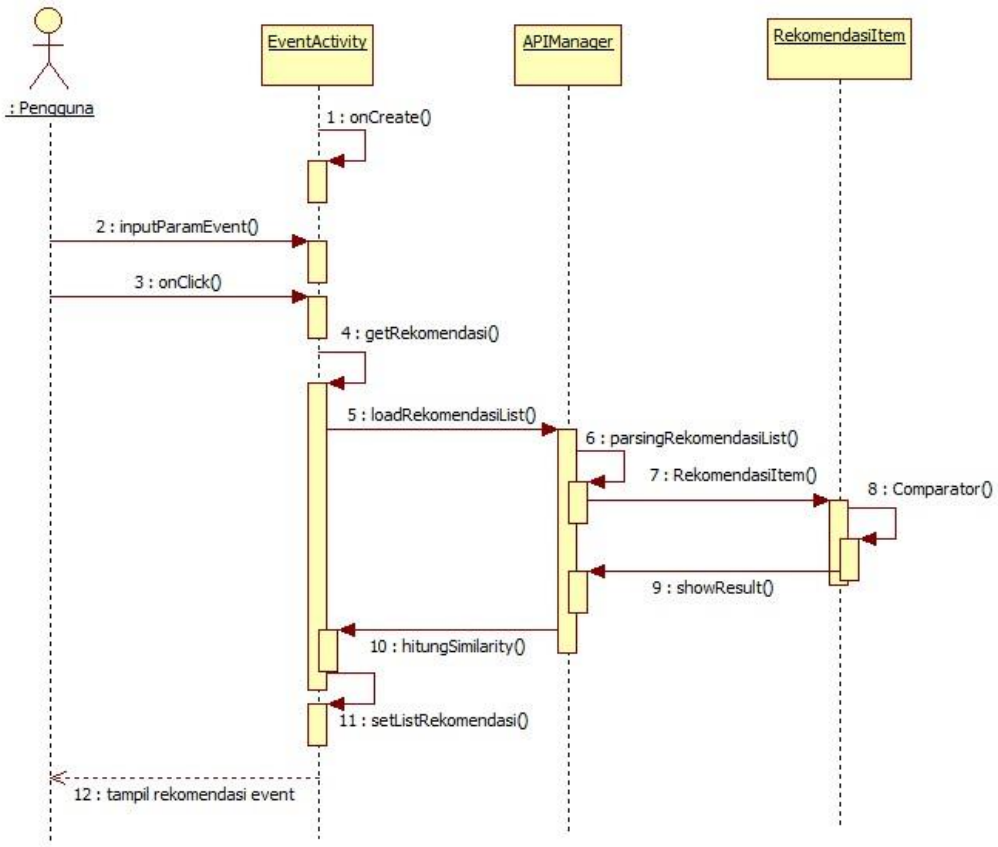
Gambar 3. 29 Class Diagram

3.1.10.5 Sequence Diagram

Sequence diagram merupakan gambaran interaksi masing-masing class dan objek pada setiap *use case* dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian data antar *class* yg di dalamnya berisi objek-objek yang saling berinteraksi. Berikut ini adalah *Sequence* Diagram yang dapat digambarkan, diantaranya adalah :

1. Sequence Diagram Melihat Rekomendasi OOTD Berdasarkan Event

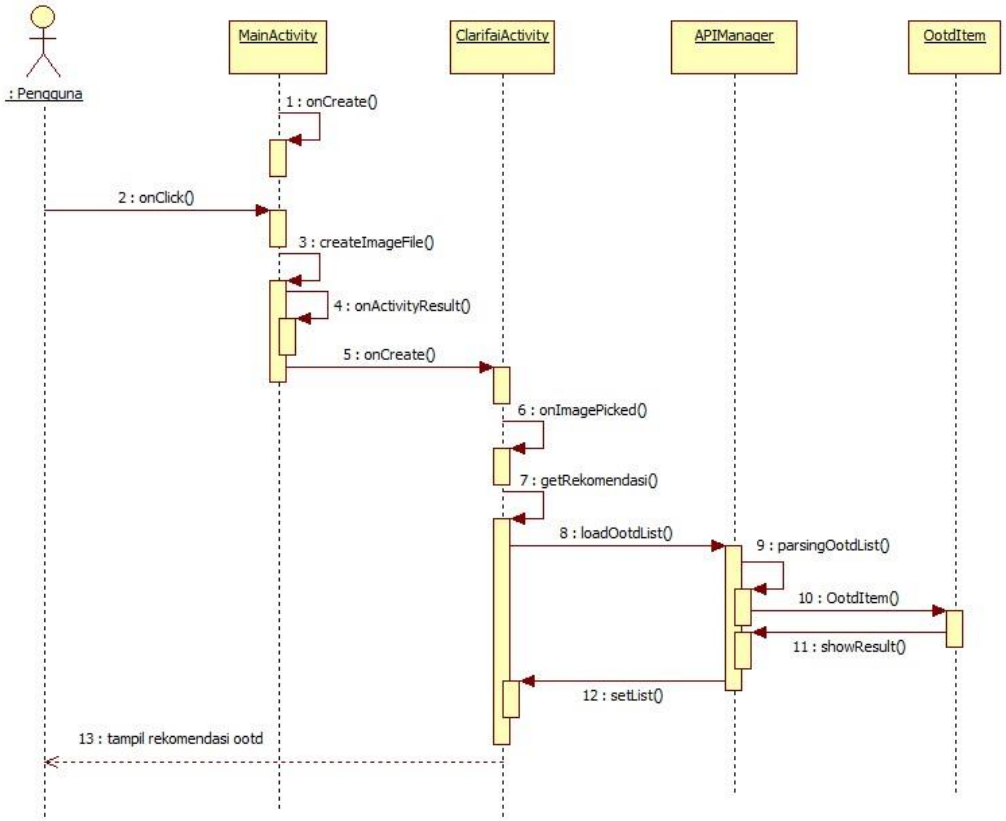
Sequence diagram melihat rekomendasi ootd berdasarkan *event* dapat dilihat pada gambar sebagai berikut :



Gambar 3. 30 Sequence Diagram Melihat Rekomendasi OOTD Berdasarkan Event

2. Sequence Diagram Melihat Rekomendasi OOTD Berdasarkan Mix and Match Item

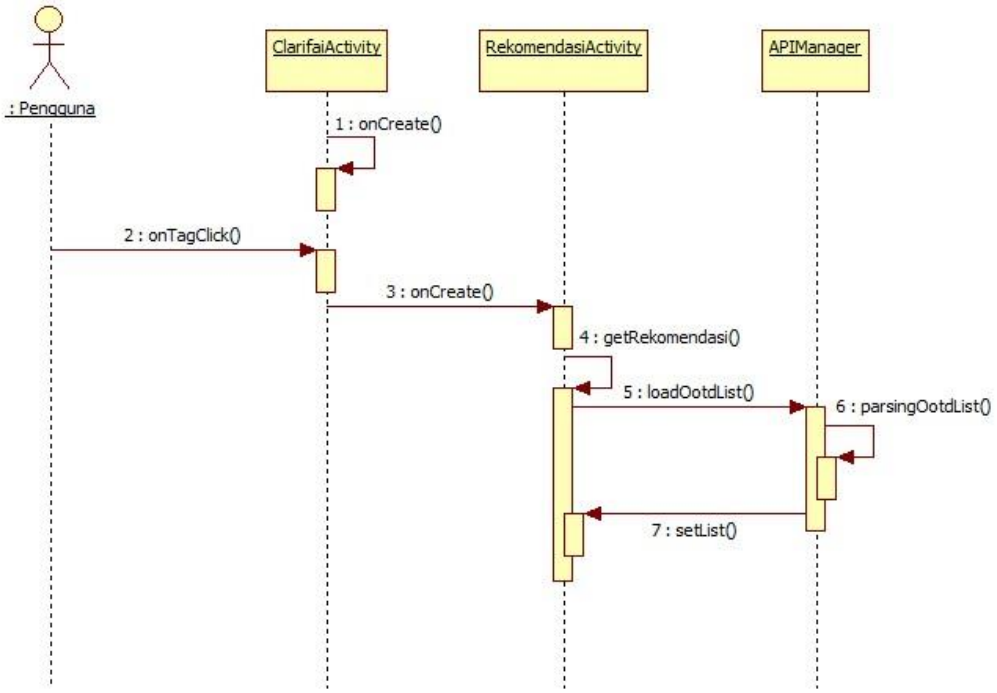
Sequence diagram melihat rekomendasi ootd berdasarkan *mix and match item* dapat dilihat pada gambar sebagai berikut :



Gambar 3. 31 Sequence Diagram Melihat Rekomendasi OOTD Berdasarkan Mix and Match Item

3. Sequence Diagram Melihat Hasil Tag

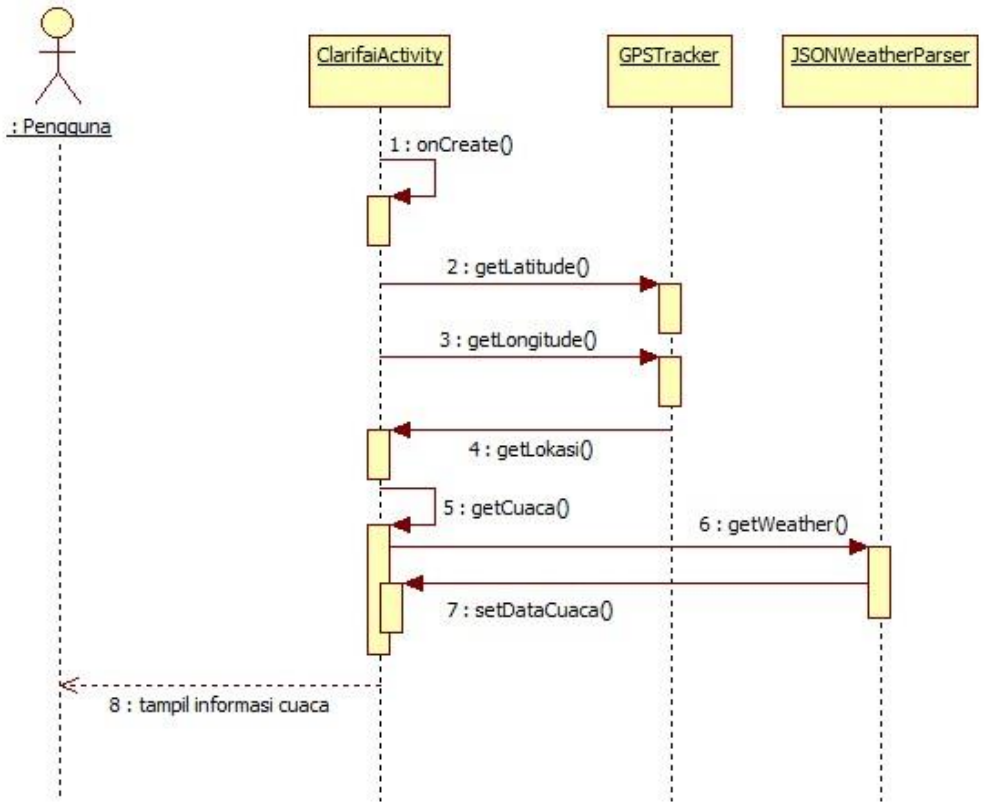
Sequence diagram melihat hasil tag dapat dilihat pada gambar sebagai berikut :



Gambar 3. 32 Sequence Diagram Melihat Hasil Tag

4. Sequence Diagram Melihat Informasi Cuaca

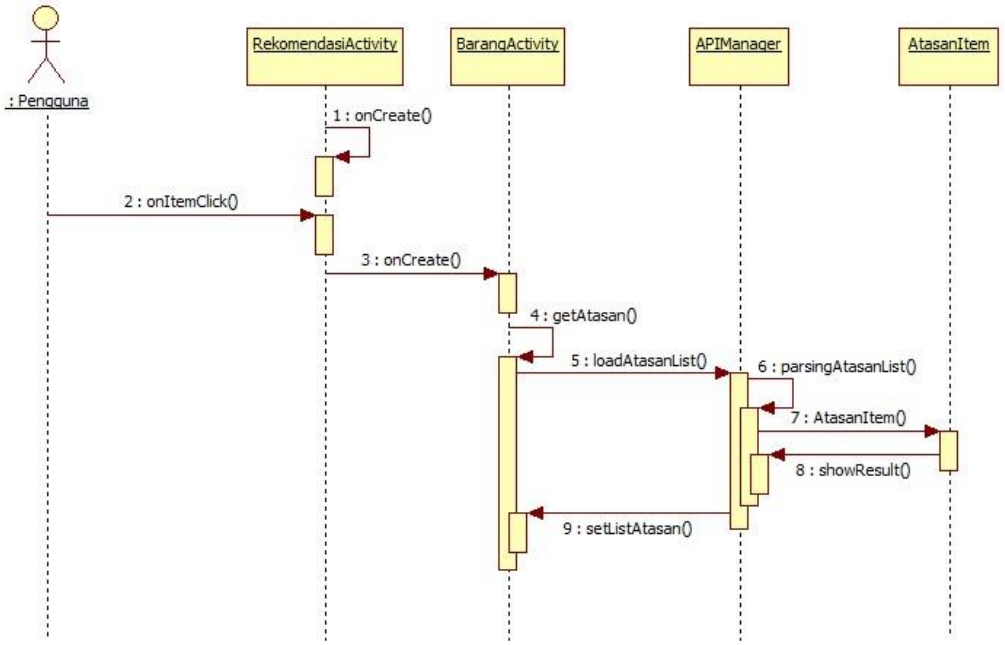
Sequence diagram melihat informasi cuaca dapat dilihat pada gambar sebagai berikut :



Gambar 3. 33 Sequence Diagram Melihat Informasi Cuaca

5. Sequence Diagram Melihat Detail Item

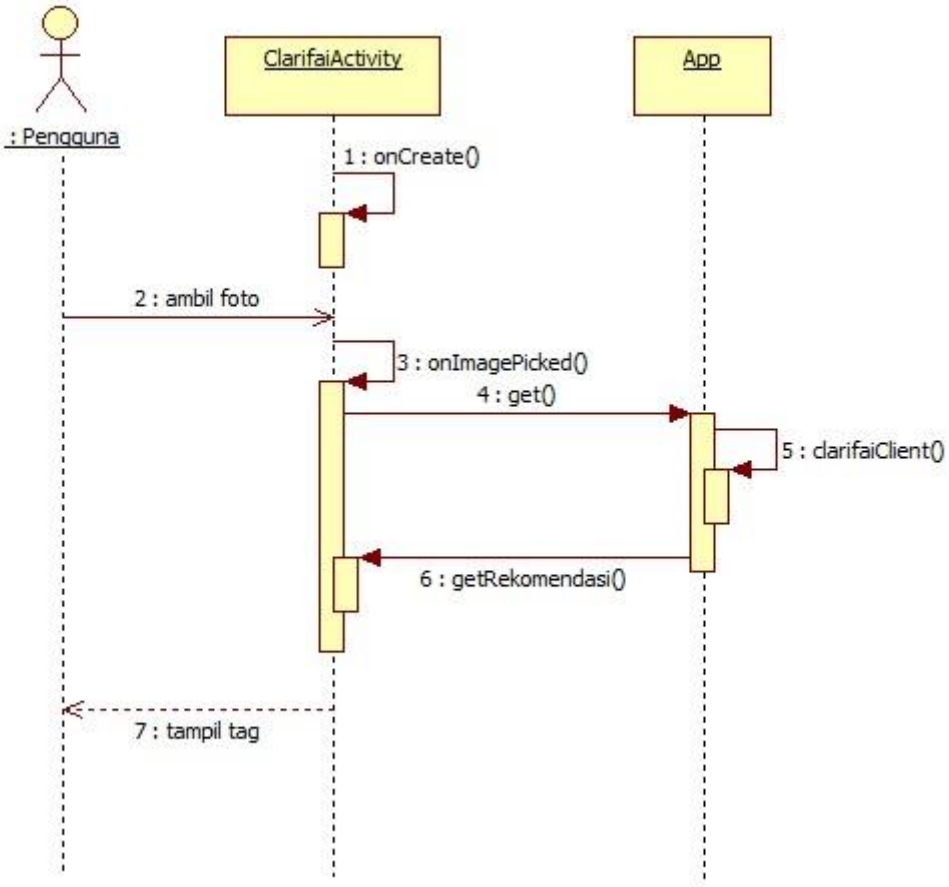
Sequence diagram melihat detail *item* dapat dilihat pada gambar sebagai berikut :



Gambar 3. 34 Sequence Diagram Melihat Detail Item

6. Sequence Diagram Memanggil URL Clarifai API

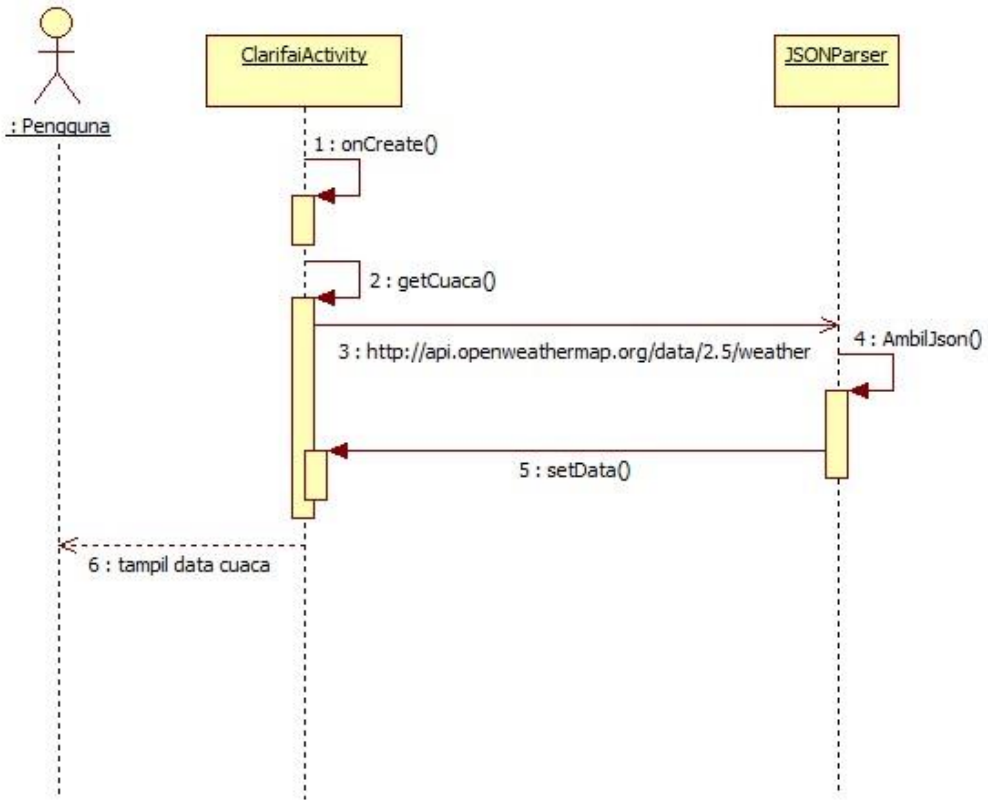
Sequence diagram memanggil URL Clarifai API dapat dilihat pada gambar sebagai berikut :



Gambar 3. 35 Sequence Diagram Memanggil URL Clarifai API

7. Sequence Diagram Memanggil URL Open Weather API

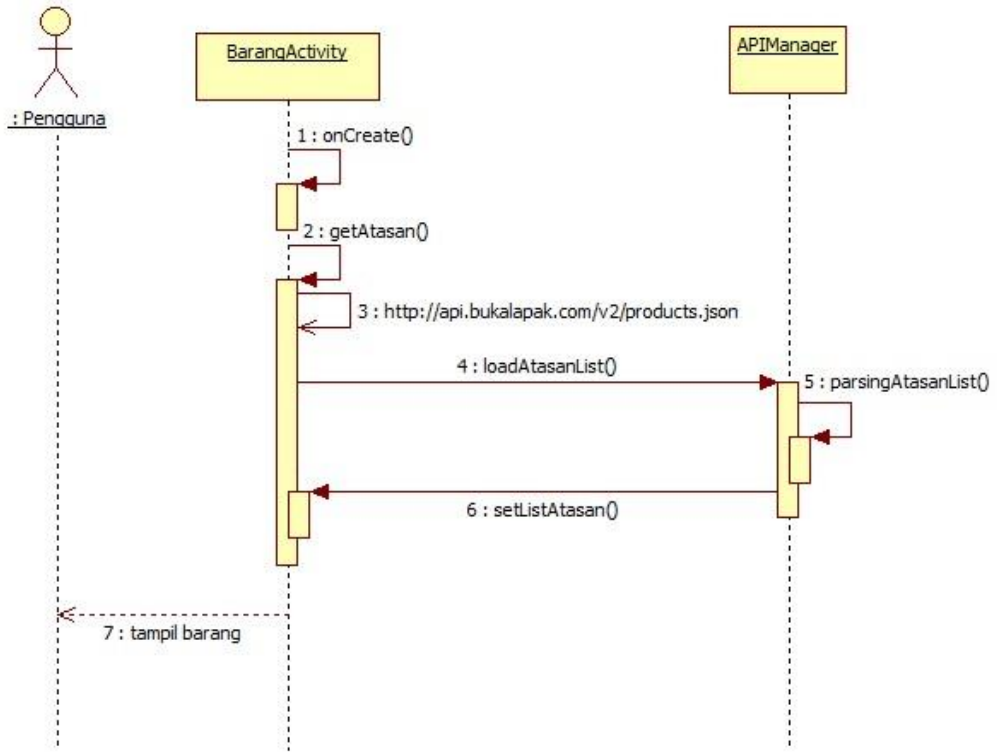
Sequence diagram memanggil URL Open Weather API dapat dilihat pada gambar sebagai berikut :



Gambar 3. 36 Memanggil URL Open Weather API

8. Sequence Diagram Memanggil URL Toko Online API

Sequence diagram memanggil URL Toko *online* API dapat dilihat pada gambar sebagai berikut :



Gambar 3. 37 Sequence Diagram Memanggil URL Toko Online API

3.2 Perancangan Sistem

Perancangan sistem merupakan penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terdapat dalam penggunaan perangkat lunak.

3.2.1 Perancangan Request dan Response Web Service

Request : http://api.bukalapak.com/v2/products.json?keywords= "+ atasan +" &category_id ="+id_jk+"&per_page=50

Response :

parameter = atasan & categori_id

response
= {"status":"OK","promo_due":null,"banner":null,"products":[{"deal_info":{},"deal_request_state":"can
request","price":54000,"category_id":14,"category":"Dress","category_structure":["Fashion
Wanita","Dress"],"courier":["JNE REG","JNE YES","J&T
REG"],"seller_username":"pradaferry","seller_name":"pradashop","seller_id":10801359,"seller_avatar":"https://s1.bukalapak.com/avt/66232701/medium/pradashop%20pusat%20aksesoris%20HP%20murah%20berkualitas.png","seller_level":"BL
User","seller_level_badge_url":"https://www.bukalapak.com/images/badge/seller/xhdpi/level-1.png</p>\n","seller_alert":null,"for_sale":true,"state_description":[],"premium_account":true,"top_merchant":false,"last_order_schedule":{"1":"16:30","2":"16:30","3":"16:30","4":"16:30","5":"16:30","6":"16:30"},"seller_voucher":{},"waiting_payment":0,"sold_count":0,"specs":{"Brand":"Lainnya","ukuran":["L",""]},"force_insurance":false,"free_shipping_coverage":[],"labels":[{"id":3495446,"name":"pakaian","slug":"pakaian","description":null}],"id":"8ebotp","url":"https://www.bukalapak.com/p/fashion-wanita/dress/8ebotp-jual-midi-dress","name":"Midi Dress","active":true,"city":"Solo","province":"Jawa
Tengah","weight":500,"image_ids":[1238851681],"images":["https://s1.bukalapak.com/img/1861588321/large/Midi_Dress.jpg"],"small_images":["https://s1.bukalapak.com/img/1861588321/small/Midi_Dress.jpg"],"desc":"Mini Dress
Mat : Peach Sofie
LD 100
Warna : Army, Mustard, Dusty","condition":"new","stock":10,"favorited":false,"created_at":"2017-05-18T13:28:17.000+07:00","updated_at":"2017-05-19T09:04:02.000+07:00","product_sin":[],"rating":{"average_rate":0,"user_count":0},"current_variant_name":"","current_product_sku_id":499991511,"product_sku":[],"options":[],"interest_count":0,"last_relist_at":"2017-05-18T13:28:17.000+07:00","view_count":8}],"categories":[],"message":null,"facets":[],"labels":null,"related_keywords":null,"recommended_products":[],"product_deeplink":null}

Request : https://www.instagram.com/explore/tags/ootd/?__a=1

Response :

{"tag": {"name": "ootd", "content_advisory": null, "media": {"nodes": [{"comments_disabled": false, "id": "1570705856747818048", "dimensions": {"height": 1350, "width": 1080}, "owner": {"id": "33620884"}, "thumbnail_src":

"https://scontent-sit4-1.cdninstagram.com/t51.2885-15/s640x640/sh0.08/e35/c0.135.1080.1080/20482012_1930697537184368_886593047893114880_n.jpg", "thumbnail_resources": [], "is_video": false, "code": "BXMRKeeFExA", "date": 1501462750, "display_src": "https://scontent-sit4-1.cdninstagram.com/t51.2885-15/e35/20482012_1930697537184368_886593047893114880_n.jpg", "caption": "Just did 85 min at the gym after eating my way through Maui for a week. Can I please go back there?!?! \ud83d\ude0d\ud83c\udf34\ud83c\udf34\ud83c\udf34\n#maui #mauiwowie #tropical #lei #flowercrowns #islandlife #island #vacation #makeup #cosmetics #skincare #freckles #ootd #wiwt #model #gym #fitness #weekend #summer #lifestyle #lifestyleblogger #la", "comments": {"count": 0}, "likes": {"count": 0}}, {"comments_disabled": false, "id": "1570705841270934708", "dimensions": {"height": 1350, "width": 1080}, "owner": {"id": "1918902483"}, "thumbnail_src": "https://scontent-sit4-1.cdninstagram.com/t51.2885-15/s640x640/sh0.08/e35/c0.135.1080.1080/20479392_154367151807540_8605185101745094656_n.jpg", "thumbnail_resources": [], "is_video": false, "code": "BXMRKQDlcy0", "date": 1501462749, "display_src": "https://scontent-sit4-1.cdninstagram.com/t51.2885-15/e35/20479392_154367151807540_8605185101745094656_n.jpg", "caption": "TENNE-HANDCRAFTED-MODERN \ud83d\udc9b\n.\n.\n.\n\u8cfc\u5165\u3002\n\u697d\u3057\u307f\u301c\u301c\ud83d\udc9b\n.\n.\n.\n#tennehandcraftedmodern #\u65b0\u4f5c#\u30aa\u30fc\u30eb\u30a4\u30f3\u30ef\u30f3#ocaille#\u8cfc\u5165#\u30b2\u30c3\u30c8\u3057\u305f #\u7740\u304f\u306e\u697d\u3057\u307f#\u305a\u3063\u3068\u697d\u3057\u307f\u306b\u3057\u3066\u305f#feelingood #fashion #vscocam #ootd", "comments": {"count": 0}, "likes": {"count": 1}}, {"comments_disabled": false, "id": "1570705827245091187", "dimensions": {"height": 1080, "width": 1080}, "owner": {"id": "3272406236"}, "thumbnail_src": "https://scontent-sit4-1.cdninstagram.com/t51.2885-15/s640x640/sh0.08/e35/20398218_566582703731544_1603170515453739008_n.jpg", "thumbnail_resources": [], "is_video": false, "code": "BXMRKC_lGlz", "date": 1501462747, "display_src": "https://scontent-sit4-1.cdninstagram.com/t51.2885-15/e35/20398218_566582703731544_1603170515453739008_n.jpg", "caption": "Getcho Shine On\u2728zoom for details:: Vintage Tee Michael Jackson 88 \"BAD\" Tour concert tee from @offthewave85 \u2022\ud83d\udcf8 @itsarielg \ud83d\udd25\n\ud83d\udd25\n\ud83d\udd25\n#lit #stylist #style #super #styleblogger #nycstyle #nycstylist #nycfashion #mensfashion #menstyle #menwithstyle #summertime #millinery #vintagetee #fashion #details #ootd #model #art #designer #dapper #vintageclothing #fashionkiller #gentlemen #nyc #love #look #follow #sewing #photooftheday", "comments": {"count": 0}, "likes": {"count": 1}}, {"comments_disabled": false, "id": "1570705820567450824", "dimensions": {"height": 1080, "width": 1080}, "owner": {"id": "4630090263"}, "thumbnail_src": "https://scontent-sit4-1.cdninstagram.com/t51.2885-15/s640x640/sh0.08/e35/20398365_1950145961887740_6035839367816151040_n.jpg", "thumbnail_resources": [], "is_video": false,

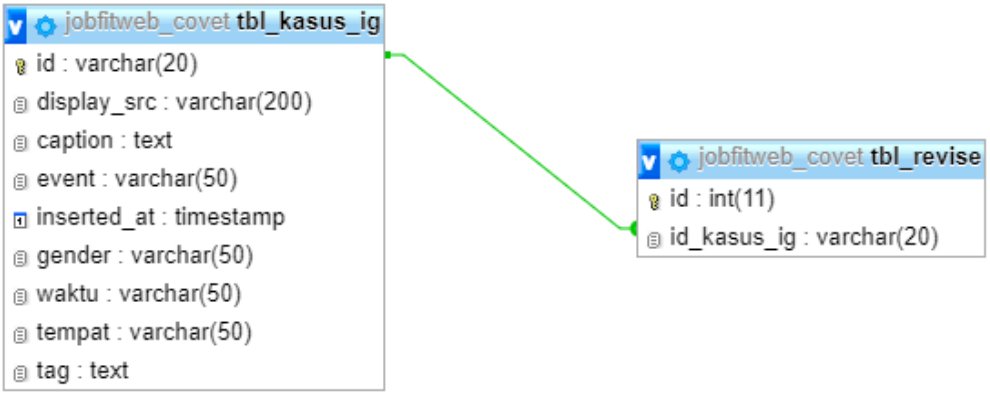
```
"code": "BXMRJ8xj7TI", "date": 1501462746, "display_src":
"https://scontent-sit4-1.cdninstagram.com/t51.2885-
15/e35/20398365_1950145961887740_6035839367816151040_n.jpg",
"caption": "PO, ready in 1 week.\nCode: 18249\nIDR
375,000\n\u2022 Embroidered denim, heavy thick material, medium
```


3.2.2 Perancangan Data

Perancangan data merupakan tahapan untuk memetakan model konseptual ke model basis data yang akan dipakai. Perancangan basis data terbagi menjadi dua yaitu skema relasi dan perancangan struktur tabel.

3.2.2.1 Skema Relasi

Skema relasi merupakan rangkaian hubungan antara dua tabel atau lebih pada sistem basis data. Skema relasi dari perangkat lunak yang dibangun adalah sebagai berikut.



Gambar 3. 38 Skema Relasi

3.2.2.2 Struktur Tabel

Struktur tabel menggambarkan detail tabel yang berisi field, tipe data, panjang data dan keterangan lainnya. Tabel yang terdapat pada basis data perangkat lunak yang akan dibangun adalah:

1. Tabel tbl_kasus_ig

   Struktur tabel tbl_kasus_ig dapat dilihat pada tabel berikut.

Tabel 3. 38 Struktur Tabel tbl_kasus_ig

Nama Field	Tipe Data	Panjang	Ekstra	Kunci
id	varchar	20	*Auto Increment*	*Primary Key*
Display_src	varchar	200		
caption	text			
event	varchar	50		
Inserted_at	timestamp			
gender	varchar	50		
waktu	varchar	50		
tempat	varchar	50		
tag	text			

2. Tabel tbl_revise

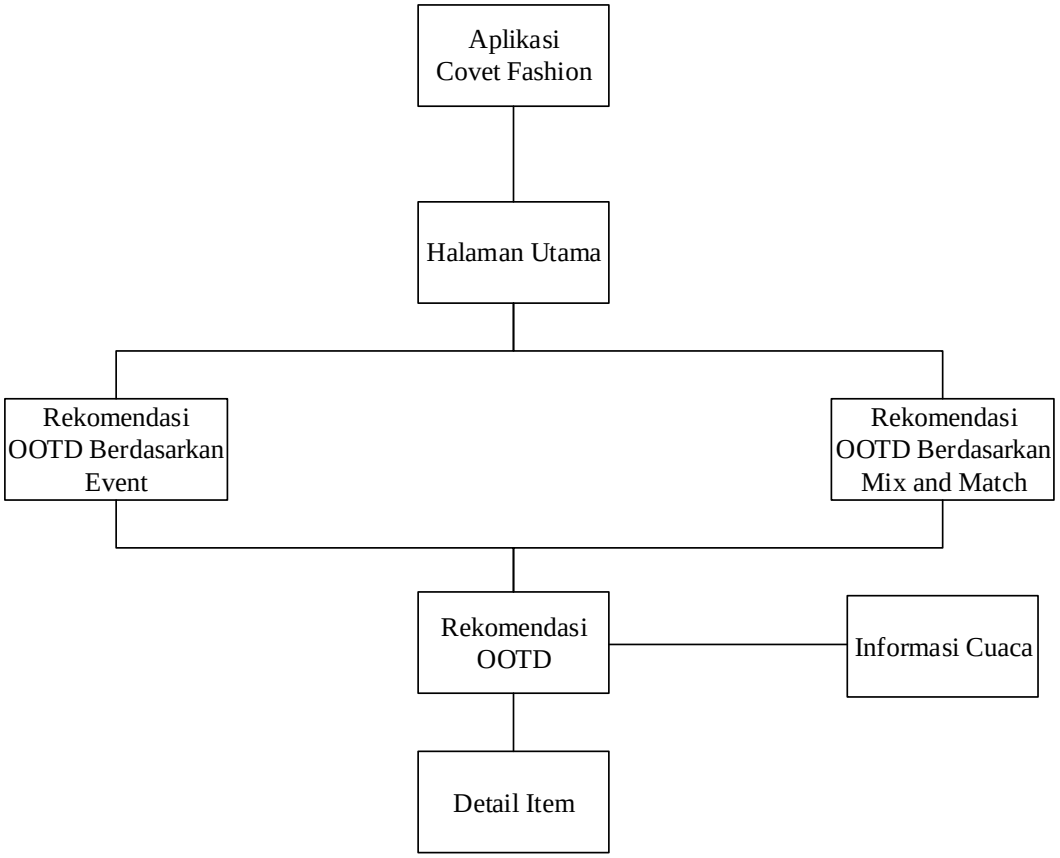
   Struktur tabel tbl_ revise dapat dilihat pada tabel berikut.

Tabel 3. 39 Struktur Tabel tbl_revise

Nama Field	Tipe Data	Panjang	Ekstra	Kunci
Id	int	11	*Auto Increment*	*Primary Key*
id_kasus_ig	varchar	20		*Foreign key* tabel tbl_kasus_ig

3.2.3 Perancangan Struktur Menu

Perancangan struktur menu merupakan gambaran jalur pemakaian aplikasi sehingga aplikasi mudah dipahami dan mudah digunakan. Perancangan struktur menu dari perangkat lunak ini adalah sebagai berikut.



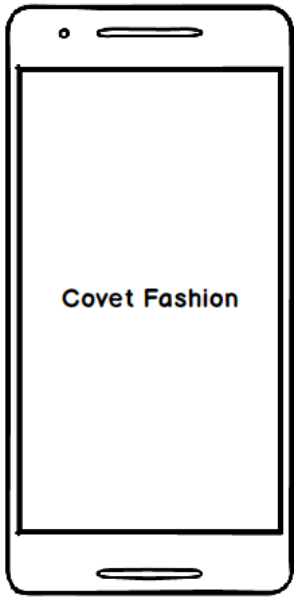
Gambar 3. 39 Struktur Menu

3.2.4 Perancangan Antarmuka

Perancangan antar muka untuk perangkat *mobile* yang akan dibangun adalah sebagai berikut:

1. Perancangan Antarmuka Splash Screen

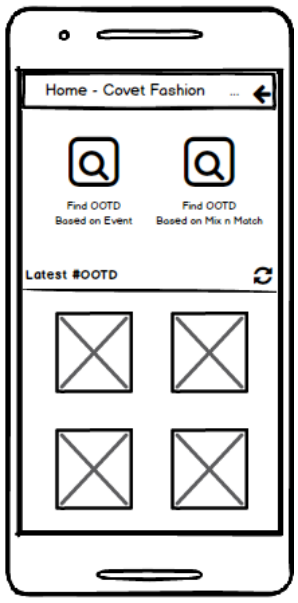
Berikut adalah perancangan yang menggambarkan rancangan antarmuka *splashscreen*

M.01 Splash Screen	Keterangan
	
Ukuran : Disesuaikan Warna font : Disesuaikan Font style : Disesuaikan Font size : Disesuaikan	

Gambar 3. 40 Antarmuka M.01 Splash Screen

2. Perancangan Antarmuka Halaman Utama

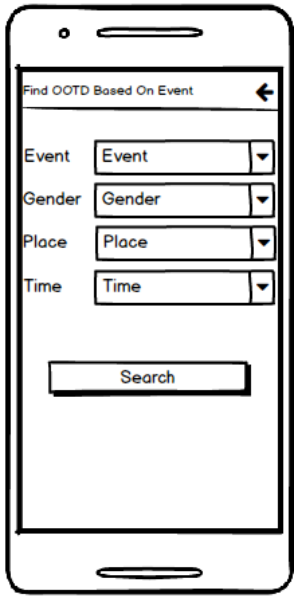
Berikut adalah perancangan yang menggambarkan rancangan antarmuka halaman utama

M.02 Halaman Utama	Keterangan
	1. Pilih menu *find* OOTD *based on event* akan menampilkan M.03 2. Pilih menu *find* OOTD *based on mix and match* akan menampilkan M.04
Ukuran : Disesuaikan Warna font : Disesuaikan Font style : Disesuaikan Font size : Disesuaikan	

Gambar 3. 41 Antarmuka M.02 Halaman Utama

3. Perancangan Antarmuka Rekomendasi OOTD Berdasarkan Event

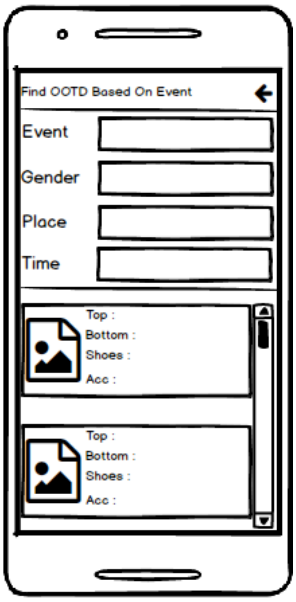
Berikut adalah perancangan yang menggambarkan rancangan antarmuka rekomendasi OOTD berdasarkan *event* :

M.03 rekomendasi OOTD berdasarkan event	Keterangan
	1. Pilih menu cari akan menampilkan M.05
Ukuran : Disesuaikan Warna font : Disesuaikan Font style : Disesuaikan Font size : Disesuaikan	

Gambar 3. 42 Antarmuka M.03 Rekomendasi OOTD Berdasarkan Event

4. Perancangan Antarmuka Hasil Rekomendasi OOTD Berdasarkan Event

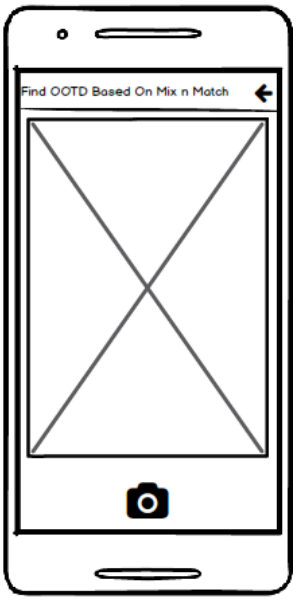
Berikut adalah perancangan yang menggambarkan rancangan antarmuka hasil rekomendasi OOTD berdasarkan *event:*

M.04 hasil rekomendasi OOTD berdasarkan event	Keterangan
	1. Klik *item* rekomendasi maka akan menampilkan M.08
Ukuran : Disesuaikan Warna font : Disesuaikan Font style : Disesuaikan Font size : Disesuaikan	

Gambar 3. 43 Antarmuka M.04 Hasil Rekomendasi OOTD Berdasarkan Event

5. Perancangan Antarmuka Rekomendasi OOTD Berdasarkan Mix and Match

Berikut adalah perancangan yang menggambarkan rancangan antarmuka rekomendasi OOTD berdasarkan *mix and match*

M.05 Menu Rekomendasi OOTD Berdasarkan Mix and Match	Keterangan
	1. Pilih menu kamera maka akan menampilkan M.06
Ukuran : Disesuaikan Warna font : Disesuaikan Font style : Disesuaikan Font size : Disesuaikan	

Gambar 3. 44 Antarmuka M.05 Rekomendasi OOTD Berdasarkan Mix and Match

6. Perancangan Antarmuka Hasil Rekomendasi OOTD Berdasarkan Mix and Match

Berikut adalah perancangan yang menggambarkan rancangan antarmuka hasil rekomendasi OOTD berdasarkan *mix 'n match*

M.06 hasil rekomendasi OOTD berdasarkan mix 'n match	Keterangan
	1. Klik menu *current weather* maka akan menampilkan M.07 2. Klik detail *item* maka akan menampilkan M.08
Ukuran : Disesuaikan Warna font : Disesuaikan Font style : Disesuaikan Font size : Disesuaikan	

Gambar 3. 45 Antarmuka M.06 Hasil Rekomendasi OOTD Berdasarkan Mix and Match

7. Perancangan Antarmuka Informasi Cuaca

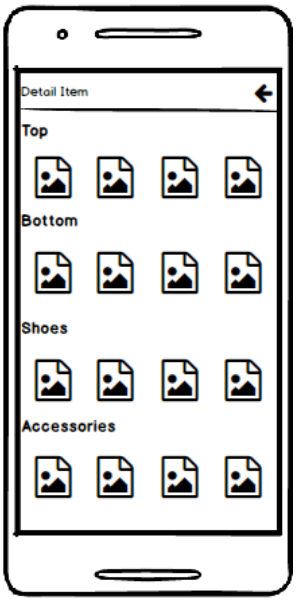
Berikut adalah perancangan yang menggambarkan rancangan antarmuka informasi cuaca

M.07 Informasi cuaca	Keterangan
	9. Klik detail *item* maka akan menampilkan M.08
Ukuran : Disesuaikan Warna font : Disesuaikan Font style : Disesuaikan Font size : Disesuaikan	

Gambar 3. 46 Antarmuka M.07 Informasi Cuaca

8. Perancangan Antarmuka Melihat Detail Item

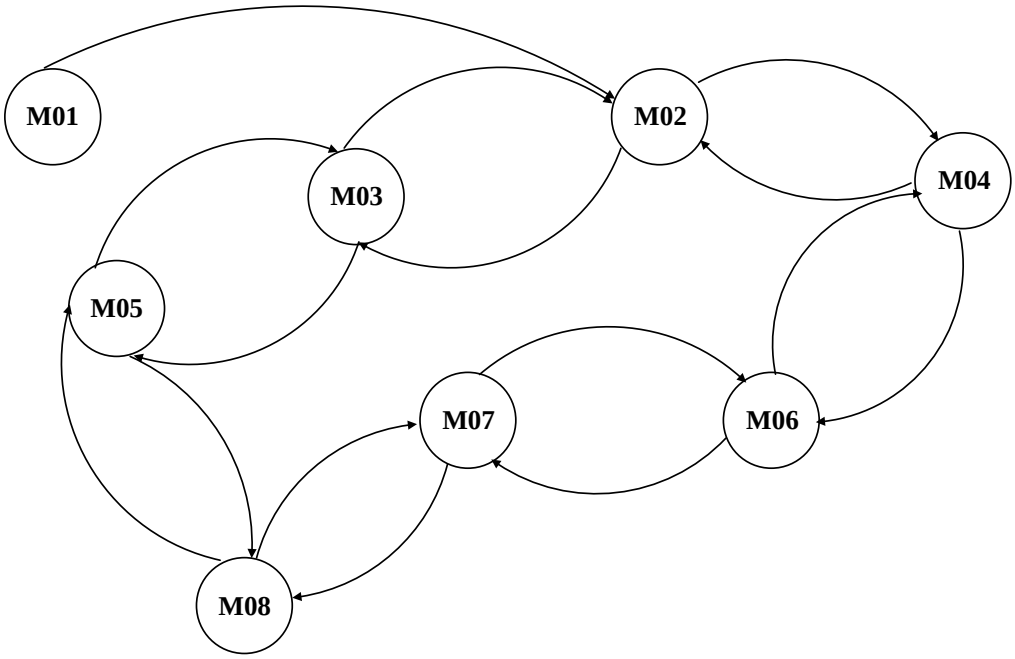
Berikut adalah perancangan yang menggambarkan rancangan antarmuka melihat detail *item*

M.08 Detail Item	Keterangan
	
Ukuran : Disesuaikan Warna font : Disesuaikan Font style : Disesuaikan Font size : Disesuaikan	

Gambar 3. 47 Antarmuka M.08 Detail Item

3.2.5 Jaringan Semantik

Jaringan Semantik merupakan gambaran pengetahuan grafis yang menunjukkan hubungan antar objek (lingkaran) dan garis yang menggambarkan formasi antar objek tersebut. Adapun jaringan semantik untuk perangkat yang dibangun adalah sebagai berikut.



Gambar 3. 48 Jaringan Semantik

Kode Tampilan	Deskripsi
M01	Halaman *splash screen*
M02	Halaman *main menu*
M03	Halaman *find OOTD based on event*
M04	Halaman *find OOTD based on mix and match*
M05	Halaman rekomendasi OOTD berdasarkan *event*
M06	Halaman rekomendasi OOTD berdasarkan *mix and match*
M07	Halaman info cuaca
M08	Detai item