BAB III
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Sistem yang dibuat merupakan informasi Jenis Kupu-kupu berbasis augmented reality yaitu pengenalan Kupu-kupu untuk pengunjung. Aplikasi yang dibuat seolah-olah pengguna dapat berinteraksi langsung dengan obyek-obyek tiga dimensi yang dibuat.

Aplikasi ini dibuat dengan mengambil latar dilingkungan nyata yang kemudian gabungkan dengan obyek-obyek 3D melalui kamera. Orientasi dan posisi Kupu-kupu yang di air keras akan dideteksi dan ditangkap oleh kamera. Setelah terdeteksi oleh kamera, maka akan didapatkan matriks transformasi yang dapat digunakan untuk transformasi seluruh obyek yang ada dalam aplikasi.

3.2 Analisis Masalah

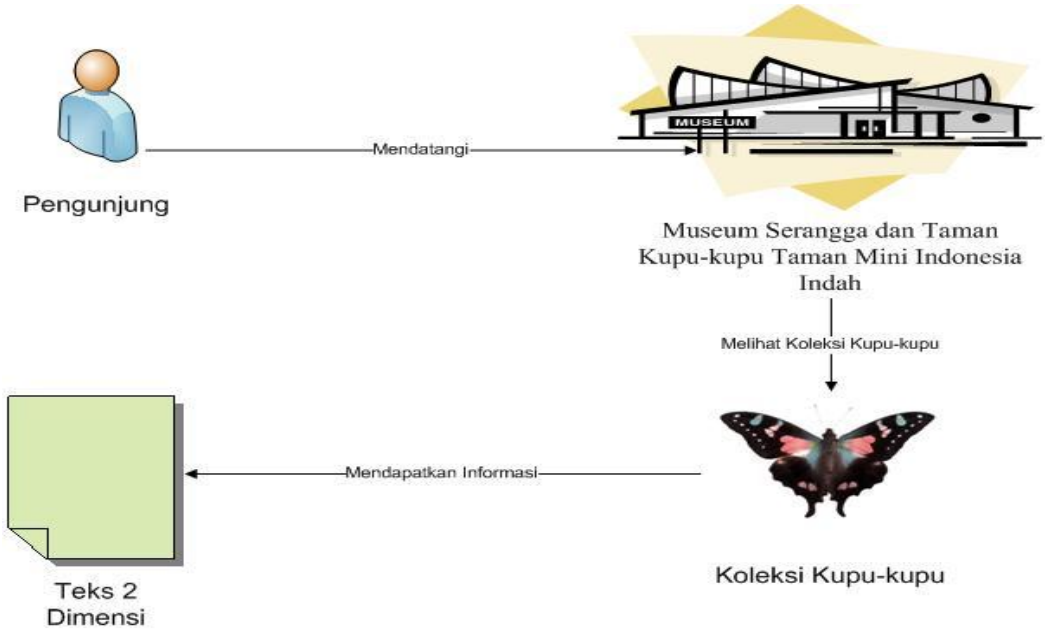
Semakin berkembangnya teknologi informasi sebagai sarana pemenuhan kebutuhan informasi bagi masyarakat dan penyampaian informasi dengan berupa teks informasi dari koleksi kupu-kupu yang di pajang, menjadi masalah yang dihadapi Museum Serangga dan Taman Kupu-kupu Taman Mini Indonesia Indah dalam beberapa tahun terakhir ini. Hal itu juga menjadi alasan bagi para pengunjung kurang mendapat informasi secara lengkap mengenai kupu-kupu yang ada. Analisis perubahan media habit dalam mengkonsumsi media di era teknologi informasi terjadi penurunan eksistensi penyampaian informasi sederhana terhadap pengunjung dengan perkembangan teknologi informasi pada saat ini.

Keunggulan dari teknologi *markerless augmented reality* dalam segi pengenalan dan menampilkan suatu informasi dapat diterapkan pada koleksi kupu-kupu untuk dapat memberikan solusi dari permasalahan yang ada. Dengan demikian, tujuan dari pembangunan aplikasi pengenalan kupu-kupu yaitu memberikan suatu aplikasi berbasis teknologi *markerless augmented reality* dalam mendeteksi kupu-kupu di Museum Serangga dan Taman Kupu-kupu Taman Mini Indonesia Indah yang telah disediakan untuk meningkatkan

penyampaian informasi secara menarik dan lengkap kepada masyarakat dengan memberikan fasilitas tambahan teknologi *pattern recognition* berupa penerapan metode deteksi pola.Dengan menerapkan teknologi ini, penyampaian informasi yang didapat akan lebih menarik dan interaktif yaitu berupa video atau animasi 3D. Sistem yang dibuat merupakan program untuk mendeteksi sebuah *markerless* sekaligus menampilkan objek tiga dimensi dan video yang bisa diputar. Objek yang di buat berupa text, animasi dan video serta menerapkan sebagian dari objek dengan metode *markerless*, agar pengunjung lebih berinteraksi dengan aplikasi dan memaksimalkan penyampaian informasi tersebut.

3.2.1 Analisis Sistem Yang Berjalan

Analisis sistem atau analisis proses adalah tahapan yang memberi gambaran tentang sistem yang sedang berjalan sekarang. Analisis ini bertujuan untuk memberi gambaran yang lebih detail bagaimana cara kerja dari sistem yang sedang berjalan. Prosedur pada proses media yang sedang berjalan sekarang dapat digambarkan ke dalam sebuah alur sistem dibawah ini.



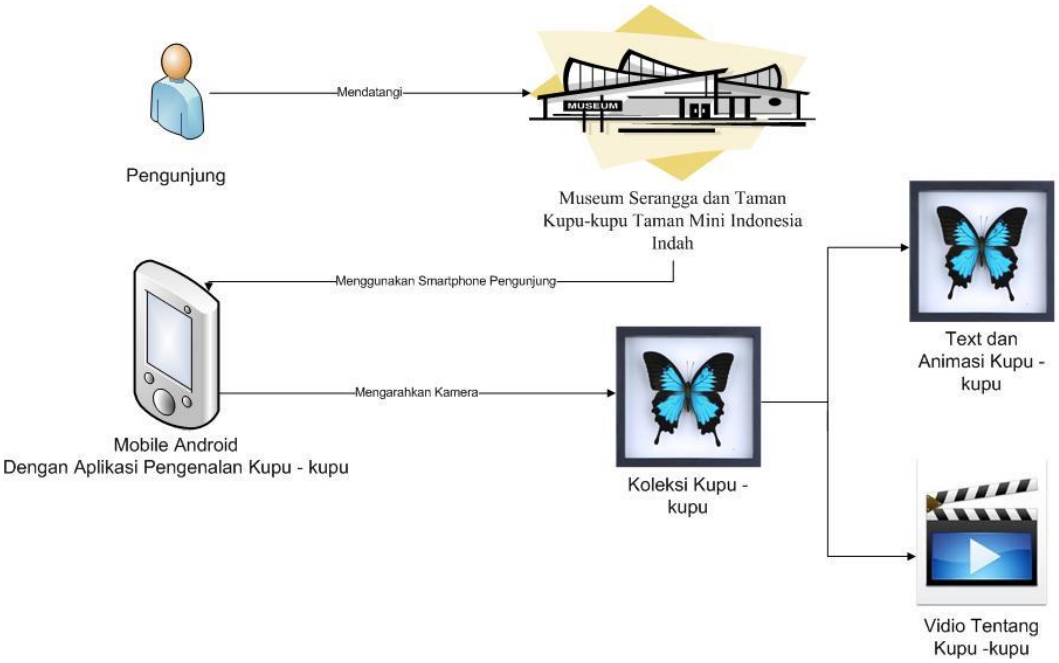
Gambar 3. 1 Alur sistem yang sedang berjalan

1. Pengunjung mendatangi Museum Serangga dan Taman Kupu-kupu Taman Mini Indonesia Indah
2. Pengunjung melihat koleksi Kupu-kupu yang dipajang di dalam Museum Serangga dan Taman Kupu-kupu Taman Mini Indonesia Indah
3. Pengunjung mendapatkan informasi Kupu-kupu berupa teks 2D

Dari gambaran prosedur yang didapat untuk mendapatkan informasi mengenai Kupu-kupu menggunakan teknologi *markerless augmented reality* bisa dijadikan sebagai media alternatif sehingga penyampaian informasi menjadi lebih menarik dan interaktif dalam bentuk sebuah alat peraga berupa *mobile* berbasis *Android*.

3.2.2 Analisis Arsitektur Sistem

Arsitektur fisik sistem yang dibangun terdiri dari komponen aplikasi Pengenalan Kupu - kupu, dan *database* aplikasi. Arsitektur sistem dalam pembangunan aplikasi ini terdapat aplikasi yang terhubung ke *database system*, menampilkan antarmuka pengunjung melalui mobile berbasis *Android*. Aplikasi ini berfungsi sebagai antarmuka antara pengunjung dengan *system database*, Pada bagian ini berisi informasi data menampilkan antarmuka pengguna dengan sistem. Fungsi utama database adalah menyimpan data dari aplikasi Pengenalan Kupu – kupu ,text,objek 3D dan *video playback*.



Gambar 3. 2 Arsitektur Sistem

Pada bagian ini, pengunjung mendatangi Museum Serangga dan Taman Kupu-kupu Taman Mini Indonesia Indah lalu *Mobile Android* dengan aplikasi Pengenalan Kupu - kupu untuk mendeteksi Kupu-kupu.Setelah itu pengunjung mengarahkan kamera ke koleksi Kupu-kupu yang telah disediakan maka muncul output berupa Text, animasi Kupu - kupu dan video sejarah Kupu-kupu. Fungsi utama Aplikasi Pengenalan Kupu-kupu adalah aplikasi pendukung yang berfungsi sebagai media pembelajaran dalam bentuk aplikasi untuk penyampaian informasi mengenai Kupu-kupu secara menarik, interaktif dan berbasis *mobile Android*.

3.2.3 Analisis Markerless

Markerless Augmented Reality merupakan salah satu metode Augmented Reality tanpa menggunakan frame marker sebagai obyek yang dideteksi. Dengan adanya *Markerless Augmented Reality* dan di dukung teknik *Pattern Recognition* (Pengenalan Pola) , maka penggunaan *marker* sebagai *tracking object* yang selama ini menghabiskan ruang, akan digantikan dengan permukaan kupu-kupu yang sudah di awetkan sebagai tracking object (obyek yang dilacak) agar dapat langsung melibatkan obyek yang dilacak tersebut sehingga dapat terlihat hidup dan interaktif, juga tidak lagi mengurangi efisiensi ruang dengan adanya marker. Terdapat perbedaan antara pelacakan berbasis *marker* (*marker based tracking*) dan pelacakan *markerless* (*markerless tracking*). Pada pelacakan berbasis *marker* posisi kamera dan orientasi kamera dhitung dengan marker yang telah ditetapkan. Sementara pelacakan *markerless*, menghitung posisi antara kamera/pengguna dan dunia nyata tanpa referensi apapun, hanya menggunakan titik-titik fitur alami (edge, corner. garis atau model 3D). Metode *Markerless* memerlukan langkah priori manual, serta model atau gambar referensi untuk inisialisasi, maka keakuratan informasi yang didapat dari object yang di traking akan lebih baik [2].

3.2.4 Analisis Algoritma Pelacak

Analisis algoritma ini menjelaskan algoritma yang digunakan vuforia dalam mendektesi object markerless yaitu Kupu-kupu

3.2.4.1 Algoritma FAST Corner detection

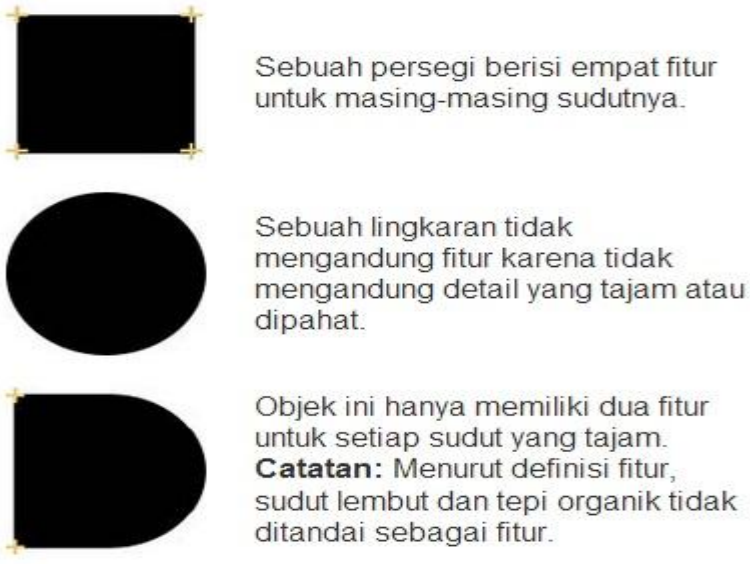
FAST *(Feture Form Accelerated segment Test)* adalah suatu algoritma yang dikembangkan oleh Edward Rosten, Reid Porter, and Tom Drummond. FAST *corner detection* ini dibuat dengan tujuan mempercepat waktu komputasi secara *real-time* dengan konsekuensi menurunkan tingkat akurasi pendeteksian sudut [19].

FAST *corner detection* dimulai dengan menentukan suatu titik p pada koordinat (x_p , y_p) pada citra dan membandingkan intensitas titik p dengan 4 titik di sekitarnya. Titik pertama terletak pada koordinat (x, y_{p-3}), titik kedua terletak

pada koordinat (x_{p+3}, y), titik ketiga terletak pada koordinat (x, y_{p+3}), dan titik keempat terletak pada koordinat (x_{p-3}, y).

Jika nilai intensitas di titik *p* bernilai lebih besar atau lebih kecil daripada intensitas sedikitnya tiga titik disekitarnya ditambah dengan suatu intensitas batas ambang (*Threshold*), maka dapat dikatakan bahwa titik *p* adalah suatu sudut. Setelah itu titik *p* akan digeser ke posisi(x_{p+1},y_p) dan melakukan intensitas keempat titik disekitarnya lagi. Iterasi ini terus dilakukan sampai semua titik pada citra sudah dibandingkan.

Vuforia menggunakan algoritma FAST Corner detection untuk mendefinisikan seberapa baik gambar dapat dideteksi dan dilacak menggunakan Vuforia SDK. Peringkat ini ditampilkan dalam Target Manager dan kembali untuk setiap target upload melalui web API. *Rating augmentabl*e dapat berkisar dari 0 sampai 5 untuk setiap gambar yang diberikan. Semakin tinggi *rating augmentable* dari target gambar, semakin kuat kemampuan deteksi dan pelacakan yang dikandungnya. Sebuah rating dari nol menunjukkan bahwa target tidak dilacak sama sekali oleh sistem Augmented Reality, sedangkan rating bintang 5 menunjukkan bahwa sebuah gambar dengan mudah dilacak oleh sistem Augmented Reality [20].



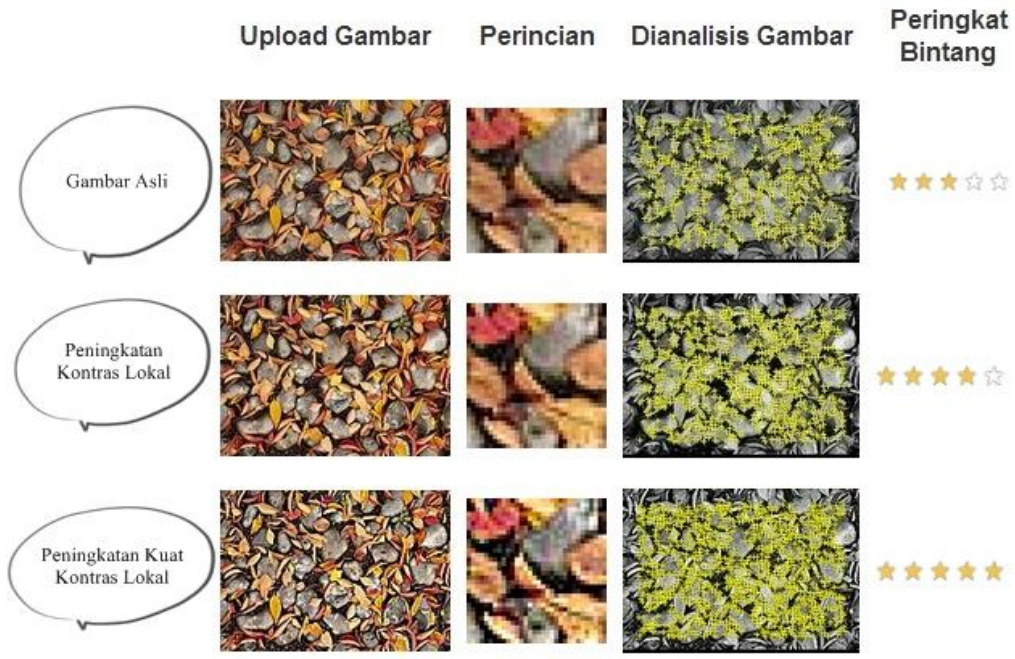
Gambar 3. 3 Contoh object yang mengandung fitur

Baik atau buruknya kontras dapat mempengaruhi terdeteksi fitur,dengan meningkatkan kontras gambar secara umum atau memilih gambar dengan detail rincian bulat, kabur dan gambar yang dikompresi berlebihan maka tidak akan memberikan kekayaan yang cukup rinci untuk dideteksi dan dilacak dengan benar.



Gambar 3. 4 Contoh rating augmentable di vuforia

Sebuah Fitur adalah tajam, berduri dan seringkali berupa sudut seperti yang hadir dalam benda berstruktur. Di bawah ini merupakan contoh struktur gambar yang memiliki fitur.



Gambar 3. 5 Contoh Perubahan kontras terhadap perubahan nilai deteksi

Gambar di atas adalah contoh yang lebih praktis tentang bagaimana untuk meningkatkan kontras lokal target. Kami menggunakan gambar dengan dua lapisan. Di latar depan adalah beberapa daun multi-warna. Latar belakang adalah permukaan bertekstur. Lapisan hanya ada dalam editor grafis kami; ketika meng-upload ke Target manajer kami selalu menggunakan gambar pipih, misalnya, format PNG. Gambar yang diunggah 512x512 piksel dalam ukuran, sedikit lebih besar dari minimum yang disarankan dari 320 piksel.

Semakin seimbang distribusi fitur dalam gambar, semakin baik gambar dapat dideteksi dan dilacak. Pertimbangkan cropping gambar untuk menghapus area tanpa fitur.



Gambar 3. 6 Contoh Cropping Gambar untuk ditribusi fitur

bentuk-bentuk organik dengan rincian lembut atau bulat yang mengandung aspek kabur atau sangat padat tidak memberikan detail yang cukup untuk dideteksi dan dilacak dengan benar atau tidak sama sekali. Bentuk tersebut mendapat count fitur rendah.



Gambar 3. 7 Contoh gambar yang tidak memberikan detail yang

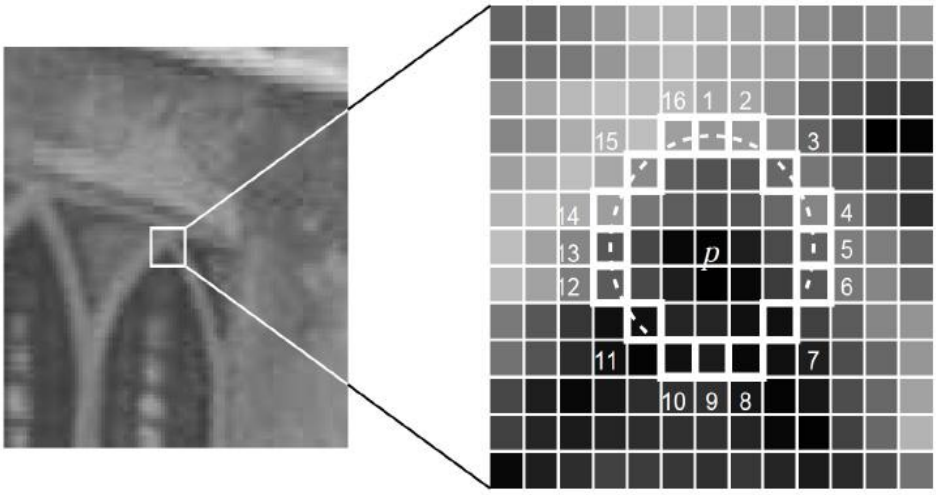
Meskipun beberapa gambar mengandung cukup fitur dan kontras yang baik, pola berulang menghambat kinerja deteksi. Untuk hasil terbaik, pilih gambar tanpa motif berulang-ulang (bahkan jika diputar dan bersisik) atau simetri rotasi yang kuat. Sebuah kotak-kotak adalah contoh dari pola berulang yang tidak dapat terdeteksi, karena pasang 2x2 kotak hitam dan putih tampak persis sama dan tidak dapat dibedakan dengan detektor.



Gambar 3. 8 Contoh gambar yang memiliki pola berulang

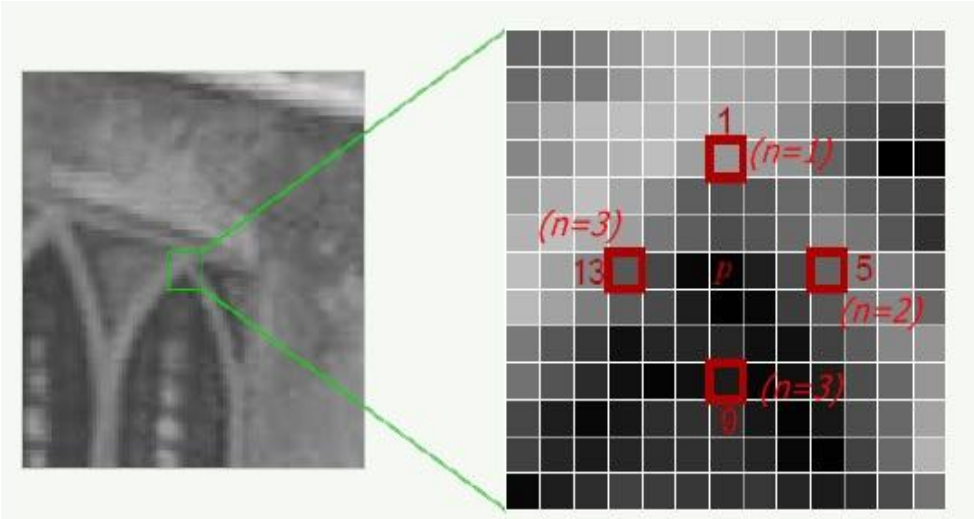
3.2.4.2 FAST corner detecteion bekerja pada suatu citra sebagai berikut:

1. Tentukan sebuah titik *p* pada citra dengan posisi awal (x_p,y_p) .



Gambar 3. 9 Gambar menampilkan titik awal yang diuji

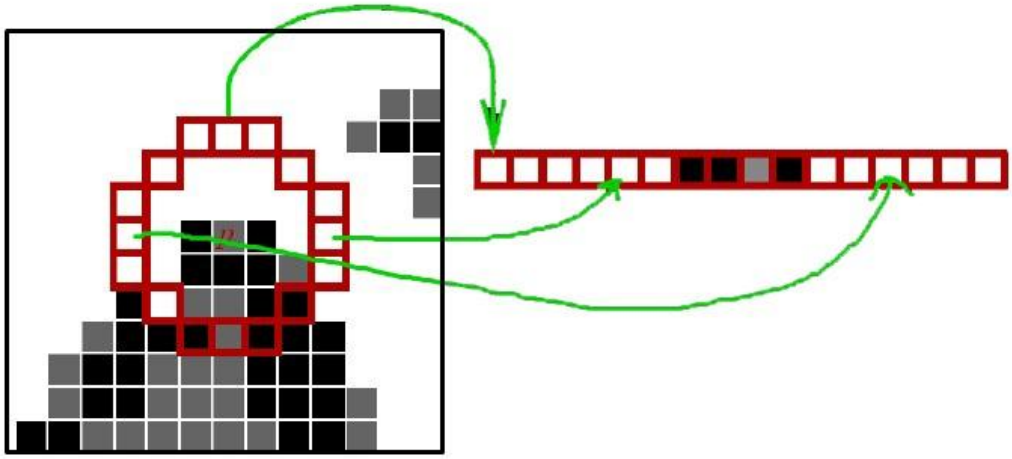
2. Tentukan keempat titik. Titik pertama (*n*=1) terletak pada koordinat (x_p,y_{p+3}), titik kedua (*n*=2) terletak pada koordinat (x_{p+3},y_p), titik ketiga terletak pada koordinat (*n*=3) terletak pada koordinat (x_p, y_{p-3}), titik keempat (*n*=4) terletak pada koordinat (X_{p-3},y_p).



Gambar 3. 10 Keempat titik koordinat

3. Bandingkan intensitas titik pusat *p* dengan keempat titik disekitar. Jika terdapat paling sedikit 3 titik yang memenuhi syarat berikut, maka titik pusat *p* adalah titik sudut.
4. Untuk menentukan titik suatu sudut, seluruh piksel akan dibagi dengan tiga subset yaitu ; Pixel dark , Pixel similiar ,dan P brighter.

5. Ulangi proses sampai seluruh titik pada citra sudah dibandingkan intensitasnya.

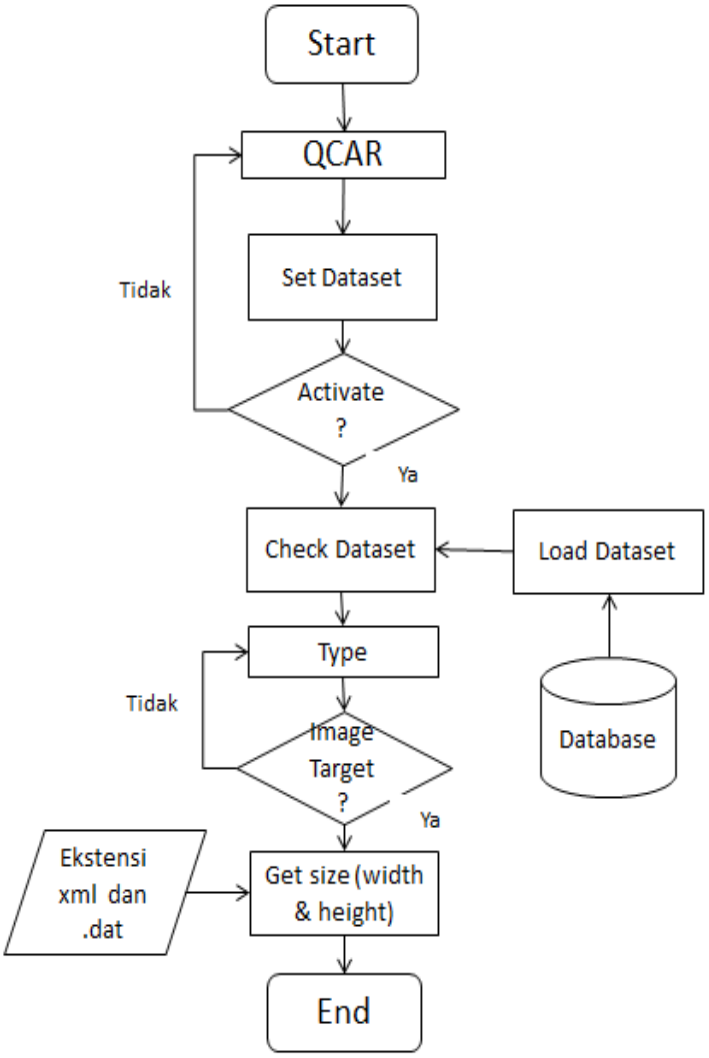


Gambar 3. 11 Tiga titik yang memenuhi syarat FAST *Corner Detection*.

3.2.5 Analisis Penentuan Markerless

Analisis metode terhadap kasus merupakan analisis yang mendeskripsikan bagaimana proses augmented reality dari awal inisialisasi, tracking penanda, sampai dengan proses memunculkan data 3D dengan metode markerless.

Dalam perancangan aplikasi dengan teknologi AR, menggabungkan objek virtual dengan objek nyata, dalam hal ini objek virtual berupa objek 3D dan objek nyatanya berupa Kupu-kupu yang ditangkap oleh kamera ponsel dengan pola tertentu(*markerless*).



Gambar 3. 12 Alur Sistem Aplikasi Augmented Reality

Secara keseluruhan, proses sistem augmented reality dapat digambarkan dengan diagram alur pada gambar 3.12. Proses pengenalan deteksi markerless ini melalui

beberapa tahapan. Secara garis besarnya, dalam perancangan ada tiga bagian utama yaitu :

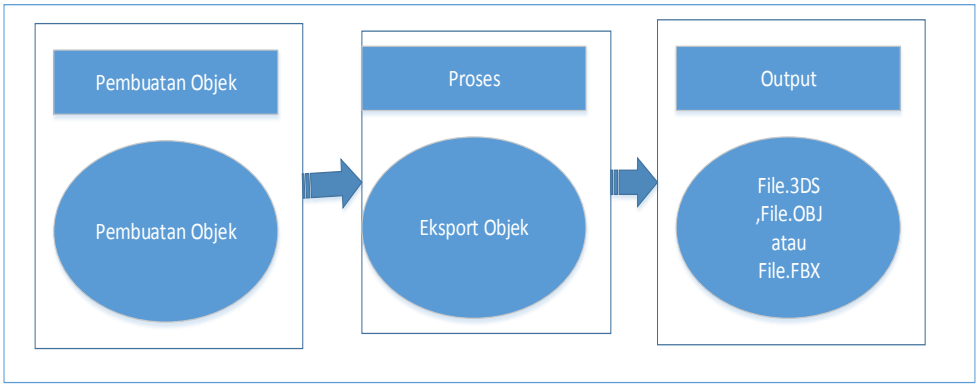
1. Inisialisasi
2. Tracking Markerless
3. Memunculkan data 3D ,Text atau Video

3.2.5.1 Inisialisasi

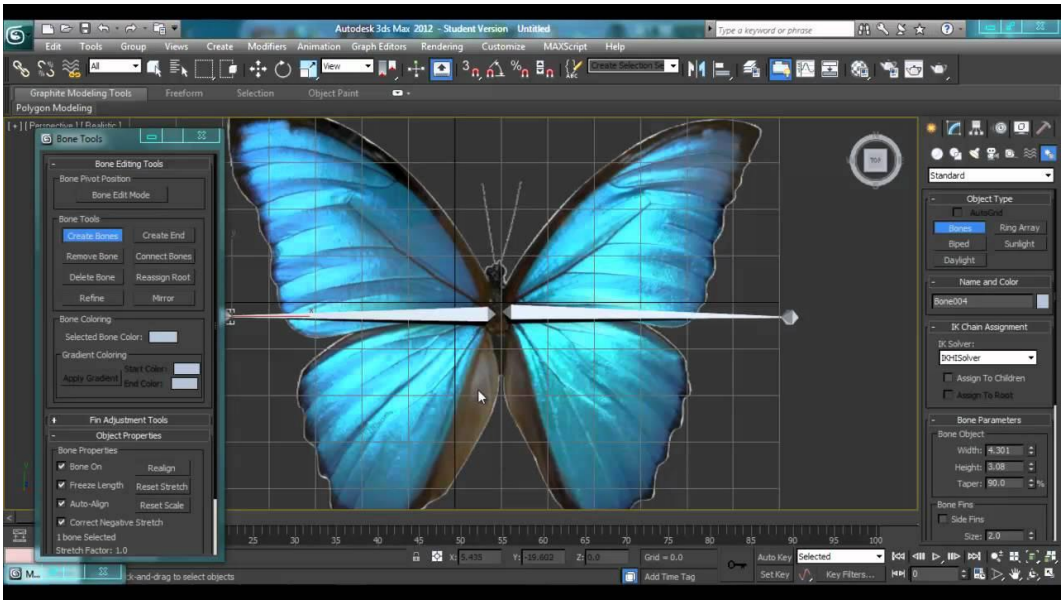
Pada tahap ini ditentukan marker yang akan digunakan, sumber input video nya, dan objek 3D yang akan digunakan .Pada bagian inisialisasi ini, objek 3D diinisialisasi terlebih dahulu karena loading objek 3D memerlukan waktu yang cukup lama. Adapun tahap inisialisasi yaitu Inisialisasi Model 3D, Inisialisasi Video , Inisialisasi pola kupu-kupu yang digunakan dan Inisialisasi Informasi Text.

3.2.5.1.1 Inisialisasi Model 3D

Model 3D yang akan ditampilkan di-load terlebih dahulu pada engine *Unity3D*. Agar aplikasi dapat menampilkan objek 3D tertentu tanpa merubah atau membangun ulang aplikasi, diperlukan sebuah file format *.3DS , .OBJ atau .FBX* untuk menentukan objek 3D yang akan di-load di Unity3D. Pembuatan objek 3D sampai dengan ringgingnya menggunakan engine 3DS Max, setelah di buat lalu diekspor ke format *.3DS , .OBJ atau .FBX* agar dapat dieksport melalui Unity3D.



Gambar 3. 13 Proses pembentukan data objek 3D



Gambar 3. 14 Proses pembuatan Model 3D pada *3DS Max*

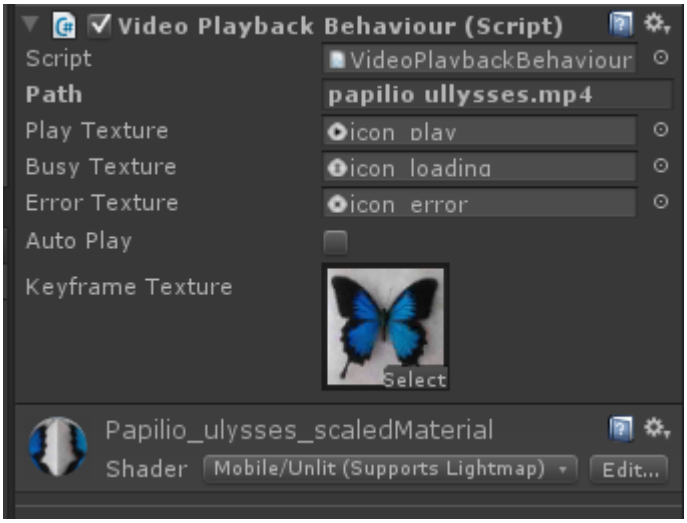
Dalam proses pembentukan objek 3D ada beberapa langkah yang dilakukan :

1. Menyesuaikan objek 3D yang dibuat dengan menambahkan tekstur yang dibutuhkan.
2. Membuat Animation jika dibutuhkan.
3. Mengekspor objek 3D yang telah dirancang menjadi format .3ds, .obj atau .fbx

3.2.5.1.2 Inisialisasi Video Player

Begitupun juga dengan Video play yang akan ditampilkan di-load terlebih dahulu di engine *Unity3D*. Agar aplikasi dapat menampilkan video tertentu tanpa merubah atau membangun ulang aplikasi.

Video ini hanya akan terlihat ketika kamera mendeteksi objek AR. Proses untuk memainkan video atau langsung diputar diperlukan menambahkan karakteristik baru ke target gambar (video) yaitu Script VideoPlaybackBehaviour.cs dan di dalamnya dapat ditemukan petunjuk untuk video dan gambar dengan ikon putar (Keyframe Tekstur).



Gambar 3. 15 Video Konfigurasi pada Unity 3D

Pada konfigurasi video diatas terdapat berberapa fungsi konfiguri yang berasal dari script VideoPlaybackBehaviour.cs , berikut penjelasannya :

1. Path : Nama dan jenis format video.
2. Play Texture : Icon Play yang akan tampil pada layar Augmented Reality jika di tekan play makan video akan mulai memutar.
3. Busy Texture: icon ini akan tampil jika perangkat dalam proses akan memuat video.
4. Error Texture: icon ini akan tampil jika video tidak dapat dimuat.
5. Auto Play : fungsi ini jika diceklis maka video akan langsung memainkan video tanpa harus disentuh icon play.
6. Key Frame Texture: tampilan frame ketika markerless objek ditemukan.

3.2.5.1.3 Inisialisasi pola Kupu-kupu Yang Digunakan

Setelah objek tiga dimensi dan video di upload maka selanjutnyanya adalah pembuatan Penanda dari objek tersebut. Penanda yang digunakan pada analisis ini didapatkan dari hasil gambar pada objek yang berada di museum serangga dan taman Kupu-kupu TMII yang sudah ada.

Proses pembuatan gambar menjadi Penanda dilakukan oleh pihak Vuforia yaitu dengan cara mengupload gambar Penanda pada sistus online developer vuforia. Setelah di convert maka dapat di unduh [21]. Setelah gambar di convert menghasilkan file dengan format .unitypackage. File tersebut kemudian dijadikan

masukan pada Engin *Unity3D* untuk mendeteksi gambar yang dijadikan Penanda *Markerless*.

Spesikasi pola Penanda :

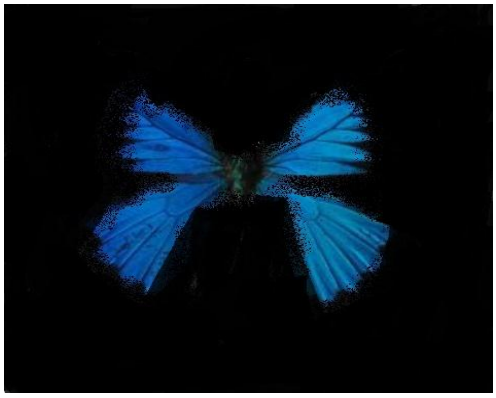
1. Pola marker minimum harus memiliki lebar 320 pixels
2. Format gambar yang dikirimkan .jpg dan .png



Gambar 3. 16 Contoh Gambar Pola Kupu-kupu

3.2.5.1.3.1 **Ambang Batas (Treshold)**

Seperti yang telah dijelaskan pada bagian analisis algoritma dalam Library Vuforia sebelum objek diconvert kedalam point-point penanda yang dapat menentukan jumlah sudut pada suatu Penanda, penanda markerless diubah kedalam bentuk Threshold sesui dengan algoritma FAST *(Feture Form Accelerated segment Test)*. Proses threshold ini yang menjadi acuan untuk proses selanjutnya, yaitu proses penentu titik markerless. Berikut contoh perubahan gambar dari gambar asli ke gambar Threshold.



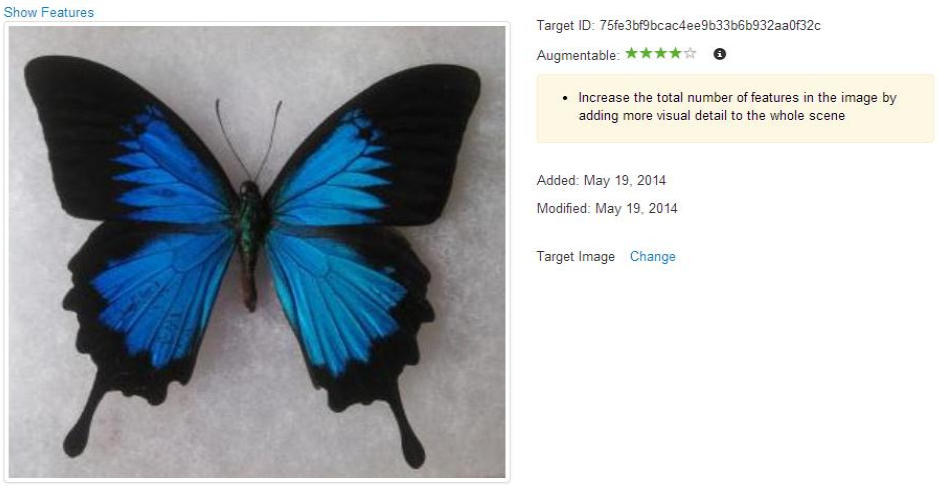
Gambar 3. 17 Gambar Treshold

Setelah menghasilkan beberapa sudut dari bentuk gambar yang telah di *Threshold* maka selanjutnya menyimpan titik point kedalam gambar *Threshold*

tersebut yang bertujuan untuk menetukan banyaknya jumlah dari titik point menggunakan fiture *Harris Corner Detection* [19].

3.2.5.1.3.2 **Fiture Harris Corener Detection**

Banyak ciri atau feature dari sebuah citra yang dapat digunakan sebagai representasi sebuah citra diantaranya adalah *Harris Corner Detection*. Fungsi dari sebuah fiture Harris ini adalah untuk menentukan titik point dari sebuah markerless. Titik sudut yang telah diciptkan akan diubah menjadi suatu titik point yang berupa bintang-bintang berwarna kuning. Semakin banyak titik point yang terdapat pada penanda makerless, maka semakin baik pola tersebut dijadikan penanda, sehingga kamera ponsel dapat dengan cepat mendeteksi penanda markerless, berikut gambar yang telah diproses menggunakan fitures *Harris Corner*.



Gambar 3. 18 Gambar Kupu-kupu yang di prosses

Setelah gambar penanda diconvert maka pola markerless didownload dan meghasilkan file.unitypackage yang didalamnya terdapat file.xml file ini lah yang berfungsi penentu untuk membedakan penanda satu dengan penanda yang lainnya.

```
<Tracking>
  <ImageTarget size="200.000000 190.926041" name="Papilio_ulysses"/>
</Tracking>
```

File.xml diatas adalah file *ImageTarget* pada *Markerless* Papilio_ulysses yang akan dipanggil oleh camera AR untuk ementukan Image Target yang akan memunculkan 3D atau Video.

3.2.5.1.4 Inisialisai Informasi Berupa Text

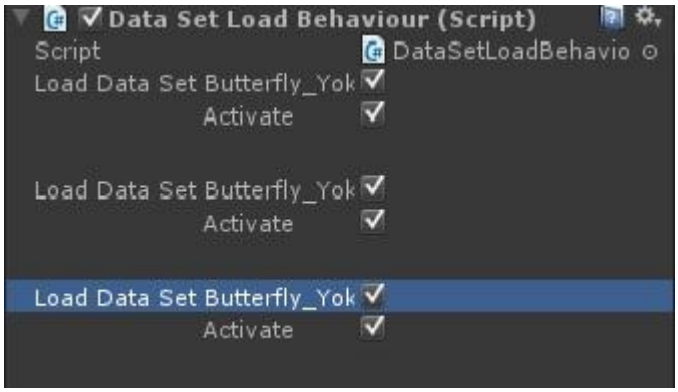
Setelah seluruh objek 3D telah di inisialisasi maka selanjutnyanya dalah menyisipkan text pada objek 3D untuk menampilkan informasi dari objek *Augmented Reality*. Penyisipan yang digunakan adalah dengan cara membuat *text gui* yang disediakan oleh engine Unity 3D , sehingga ketika objek 3D muncul maka informasi berupa text akan muncul. Hal ini berkerja hanya pada Kupu-kupu yang menapilkan bentuk realita tiga dimensi.

3.2.5.2 Tracking pola Markerless

Tracking pola Markerless adalah proses dimana mendeteksi sebuah pola yang kemudian terintegrasi untuk menghasilkan objek 3D maupun video. Didalam prosesnya setiap setiap penanda dan objek 3D beserta video akan di inisialisasikkan terlebih dahulu agar dapat mudah untuk digunakan dalam proses pencocokan pola markerless.

Proses pelacakan beberapa objek yang dapat dilacak dan diregistrasi oleh QCAR (Qualcom Augmented Reality) SDK Vuforia Unity. Dalam proses pelacakan ada beberapa parameter untuk menentukan objek yang akan dilacak. DataSetLoadBehaviour.css adalah satu dari banyaknya proses pelacakan. Kode fragmen pada script DataSetLoadBehaviour di bawah ini menggambarkan penciptaan 3 penanda yaitu : .

1. Papilio Ulysses
2. Papilio Blumei
3. Papilio Sataspes

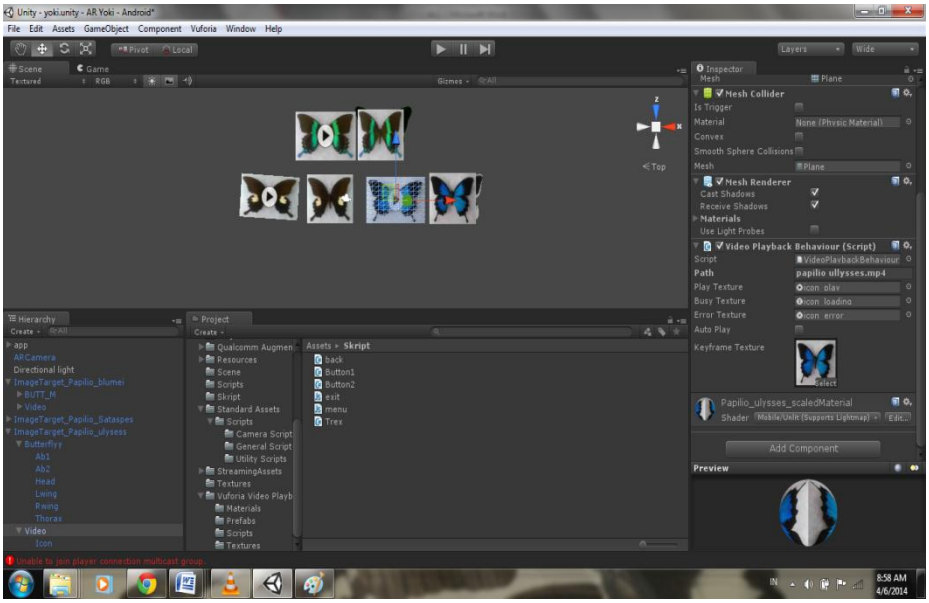


Gambar 3. 19 Script Tracking markerless

Script diatas akan disimpan pada object pembaca yaitu Camera AR yang diolah menggunakan Unity3D melalu penanda yang berformat .unitypackage.

3.2.5.3 Memunculkan data 3D dan Video

Transformasi matriks yang dikalkulasikan di step sebelumnya yang digunakan vuforia dan menampilkan objek yang sesuai dengan sebuah library 3D, seperti yang ditunjukkan gambar dibawah menyertakan kelas pendukung ke setiap kelas matriks internal library 3D tersebut. Proses pelacakan posisi dan orientasi hingga mengenali Target sebagai tempat memunculkan obyek dilakukan dengan sistem QCAR (Qualcom Augmented Reality). Sedangkan Unity 3D berperan dalam menciptakan obyek maya 3D dan proses sama seperti yang dilakukan pada lingkungan antarmuka Unity 3D. Gambar dibawah merupakan proses penciptaan 3D terhadap markerless pada Unity 3D.



Gambar 3. 20 Penciptaan 3D terhadap markerless

Dari gambar penciptaan 3D gambar diatas terlihat hasil akhir Obyek yang akan dimunculkan setelah pelacakan. Setiap obyek yang akan dimunculkan memiliki bentuk dan tekstur masing-masing. Obyek juga tidak selalu bersifat statis. Seringkali obyek yang diinginkan adalah obyek bergerak. Unity bertugas memproses ini sehingga obyek-obyek tersebut dapat muncul dengan baik pada perangkat Android.

3.2.6 Analisis Kebutuhan UNITY (Software Engine)

Unity merupakan software pembangun game. Masa kini unity dapat pula digunakan dalam pembangun aplikasi berbasis *recognition* seperti hal rupanya *augmented reality*.

Unity adalah sebuah sebuah *tool* yang terintegrasi untuk membuat *game*, arsitektur bangunan dan simulasi. Kelebihan unity yaitu *multiple platform*, maksudnya banyak platform yang disupport oleh unity, misal Windows, Mac, iPhone, iPad, Android, Nintendo Wii, dan juga browser. Untuk browser,memerlukan sebuah *plugin*, yaitu Unity Web Player, sama halnya dengan Flash Player pada Browser.

Bahasa pemrograman yang digunakan bermacam-macam, mulai dari Javascript, C#, dan Boo. Banyak hal yang bisa di lakukan di unity , ada fitur *audio reverb zone* , *particle effect* , *Sky Box* Untuk menambahkan langit , dan masih banyak lagi tentunya juga bisa langsung mengedit tekstur dari editor seperti photoshop dll.

3.2.7 Analisis

Dalam analisis sistem akan dibahas mengenai apa saja yang dibutuhkan dalam sistem aplikasi yang akan dibuat seperti analisis kebutuhan non fungsional, analisis perangkat lunak, analisis perangkat lunak untuk pengembang dan pengunjung, analisis perangkat keras pengembang dan pengunjung, analisis pengunjung, analisis fungsional, analisis kebutuhan sistem dan konten, pemodelan sistem, jaringan semantik dan perancangan method.

3.2.7.1 Analisis Kebutuhan Non Fungsional

Adapun kebutuhan non-fungsional untuk menjalankan aplikasi ini meliputi kebutuhan perangkat keras,kebutuhan perangkat lunak dan pengguna sistem yang akan memakai aplikasi. Analisis kebutuhan non-fungsional bertujuan agar aplikasi yang dibangun dapat digunakan sesuai dengan kebutuhan pengguna aplikasi dalam mencari informasi yang dibutuhkan.

3.2.7.1.1 Analisis Perangkat Lunak

Perangkat lunak digunakan dalam sebuah sistem merupakan kumpulan perintah-perintah yang diberikan kepada perangkat keras agar saling berinteraksi

untuk melakukan suatu tugas. Dalam analisis perangkat lunak ini dibagi menjadi 2 (dua) yaitu analisis perangkat lunak untuk pengembang dan analisis perangkat lunak untuk pengunjung.

3.2.7.1.2 Analisis Perangkat Lunak untuk Pengembang

Perangkat lunak yang dibutuhkan untuk membangun aplikasi Pengenalan Kupu-kupu untuk pengembang adalah sebagai berikut :

1) Sistem operasi Windows 7 Ultimate
2) Unity 4.2.1
3) Vuforia sdk
4) Autodesk 3DMax
5) Windows Movie Maker

3.2.7.1.3 Analisis Perangkat Lunak untuk Pengunjung

Perangkat lunak yang dibutuhkan untuk membangun aplikasi pengenalan Kupu-kupu untuk pengunjung adalah sebagai berikut :

1) Sistem Operasi Android
2) *Sistem aplikasi pengenalan Kupu-kupu.apk*

3.2.7.2 Analisis Perangkat Keras

Alat peraga berupa mobile dan camera terdiri dari perangkat keras dan perangkat lunak yang saling berinteraksi. Perangkat lunak memberikan instruksi - instruksi kepada perangkat keras untuk melakukan tugas tertentu. Dalam analisis perangkat lunak ini dibagi menjadi 2 (dua) yaitu analisis perangkat lunak untuk pengembang dan analisis perangkat lunak untuk pengunjung.

3.2.7.2.1 Analisis Perangkat Keras untuk Pengembang

Dalam pembangunan aplikasi pengenalan Kupu-kupu, perangkat keras yang digunakan untuk pengembang adalah sebagai berikut :

1) Processor Intel Core i5 2430M2,4Ghz
2) RAM 4GB
3) Harddisk 750GB
4) Mouse dan Keyboard
5) Webcam 1.3 Mega Pixel
6) Printer Canon IP2770
7) Handphone berbasis Android

3.2.7.2.2 Analisis Perangkat Keras untuk Pengunjung

Dalam pembangunan aplikasi pengenalan Kupu-kupu, perangkat keras yang digunakan untuk pengunjungadalah sebagai berikut :

1) *Smartphone* pengunjung
2) Koleksi Kupu-kupu di Museum Serangga dan Taman Kupu-kupu Taman Mini Indonesia Indah

3.2.7.3 Analisis Pengguna

Aplikasi deteksi jenis Kupu - kupu ini diakses oleh pengunjung maupun pemandu wisata (*guide*). Dimana pengunjung ataupun pemandu wisata (*guide*)

tersebut dapat menjalankan aplikasi ini dengan menggunakan *mobile* berbasis android lalu cameranya diarahkan ke media *target* Kupu-kupu yang sudah ditentukan.

Tabel 3. 1 Karakteristik Pengguna

Pengguna	Haks Akses	Pengalaman
User	**Mengakses data informasi melalui aplikasi**	**Minimal dapat menjalankan aplikasi pada perangkat android**

3.2.7.4 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik serta sesuai dengan kebutuhan. Analisis kebutuhan fungsional ini meliputi analisis kebutuhan sistem, analisis kebutuhan konten, dan pemodelan sistem.

3.2.7.5 Analisis Kebutuhan Sistem

Dalam pembangunan aplikasi perancangan sistem, sebelumnya dilakukan analisis kebutuhan sistem agar sistem yang dibangun sesuai dengan yang diharapkan atau dibutuhkan pengguna. Adapun kebutuhan sistem yang diperlukan antara lain :

1. Menampilkan informasi berita animasi atau video pada display.
2. Informasi Kupu-kupu animasi atau video yang diinformasikan jelas.
3. Memberikan informasi visual mengenai Kupu-kupu yang terkait dengan tampilan berupa text, animasi 3D dan video.
4. Informasi mengenai Kupu-kupu animasi atau video yang sudah diterapkan teknologi *markerless augmented reality* segera ditampilkan.

3.2.7.6 Analisis Kebutuhan Konten

Analisis ini bertujuan untuk menganalisis konten yang diperlukan agar memudahkan perancangan informasi setelah sistem yang dibutuhkan telah diketahui. Adapun analisis kebutuhan konten dengan kriteria sebagai berikut :

1. Konten Text

Menampilkan konten text berupa text informasi yang sesuai dengan Kupu-kupu.

2. Konten Animasi

   Menampilkan konten animasi berupa objek 3D.

3. Konten Video

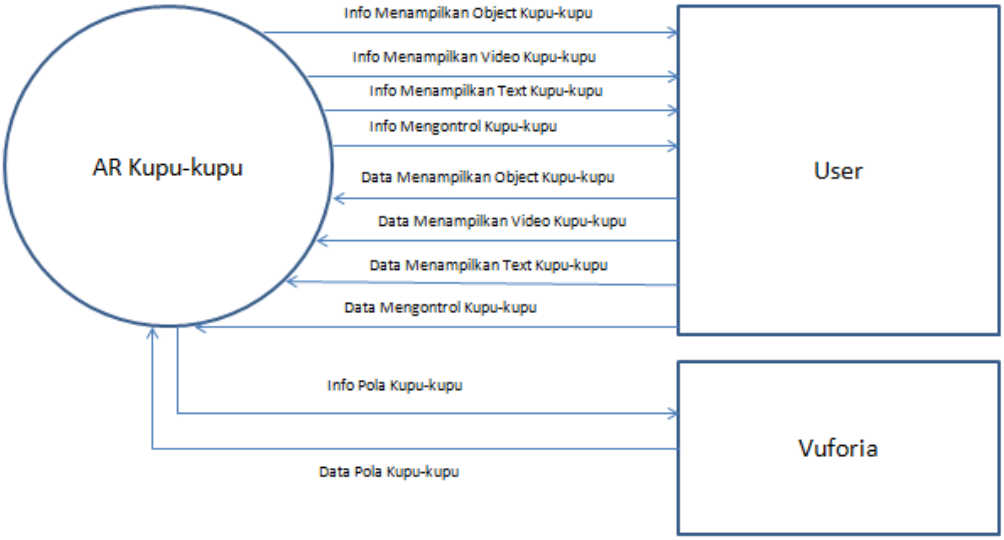
   Menampilkan konten video berupa video informasi yang sesuai dengan Kupu-kupu.

3.2.7.7 Pemodelan Sistem

Pemodelan sistem berfungsi untuk menentukan fungsi-fungsi yang dapat dilakukan oleh sistem pada aplikasi serta menentukan kelas yang dibutuhkan untuk realisasi fungsi-fungsi sistem yang telah dianalisis sebelumnya dan mendeskripsikannya kedalam bentuk diagram.

3.3 Diagram Konteks

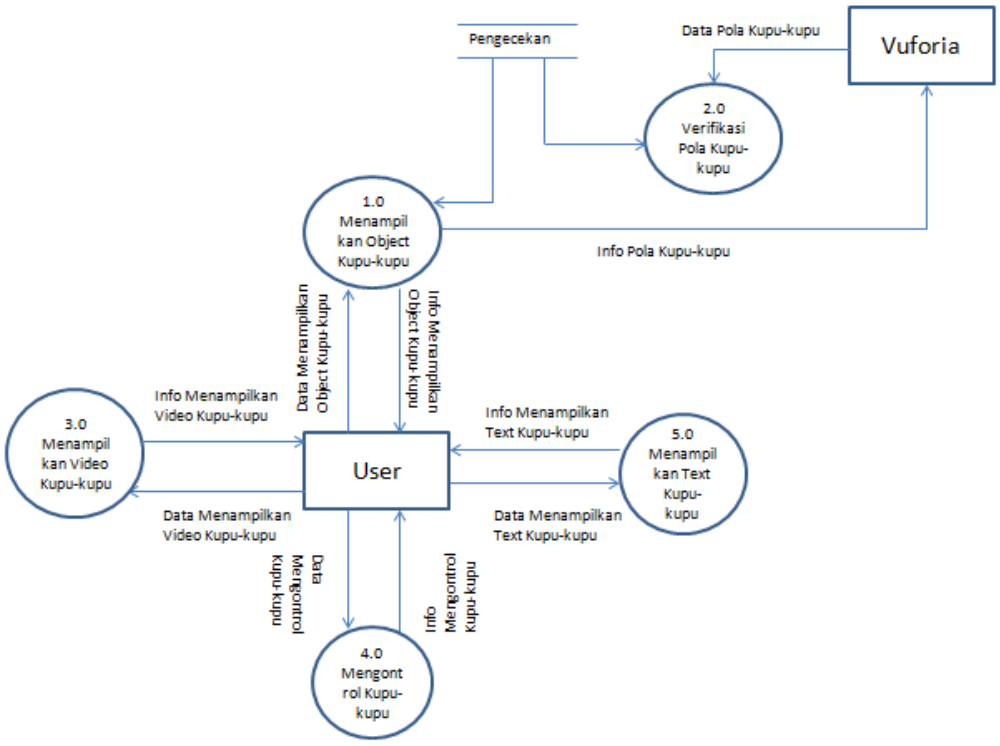
Diagram konteks adalah suatu diagram yang menggambarkan ruang lingkup laboratorium virtual atau bagaimana laboratorium virtual tersebut berinteraksi dengan lingkungannya. Sistem digambarkan dengan bulatan, sedangkan lingkungan diwakili oleh entitas luar yang digambarkan dengan persegi. Gambar 3.22 menggambarkan diagram konteks yang terdapat pada aplikasi *ini*.



Gambar 3. 21 Diagram Kontek Aplikasi

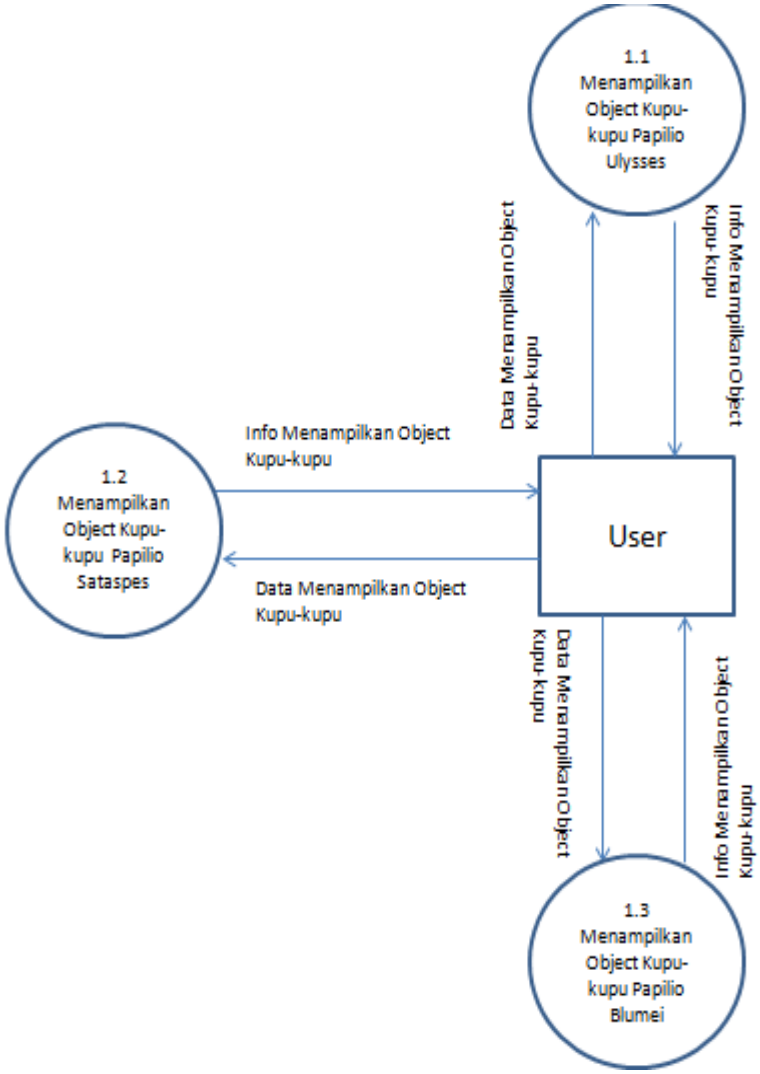
3.4 Data Flow Diagram (DFD) Level 1

Diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, tersruktur dan jelas. Gambar 3.22 menjelaskan alur dari data sistem kepada pengguna.



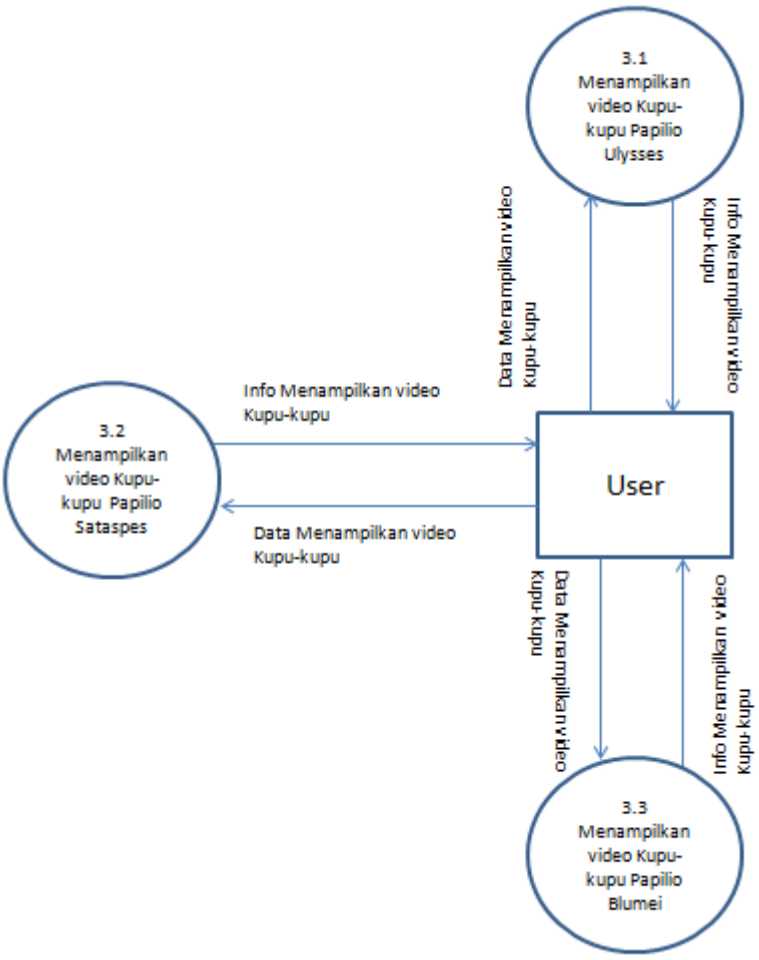
Gambar 3. 22 Data Flow Diagram Level 1

3.4.1 DFD Level 2 Proses 1.0 : Menampilkan Object Kupu-kupu



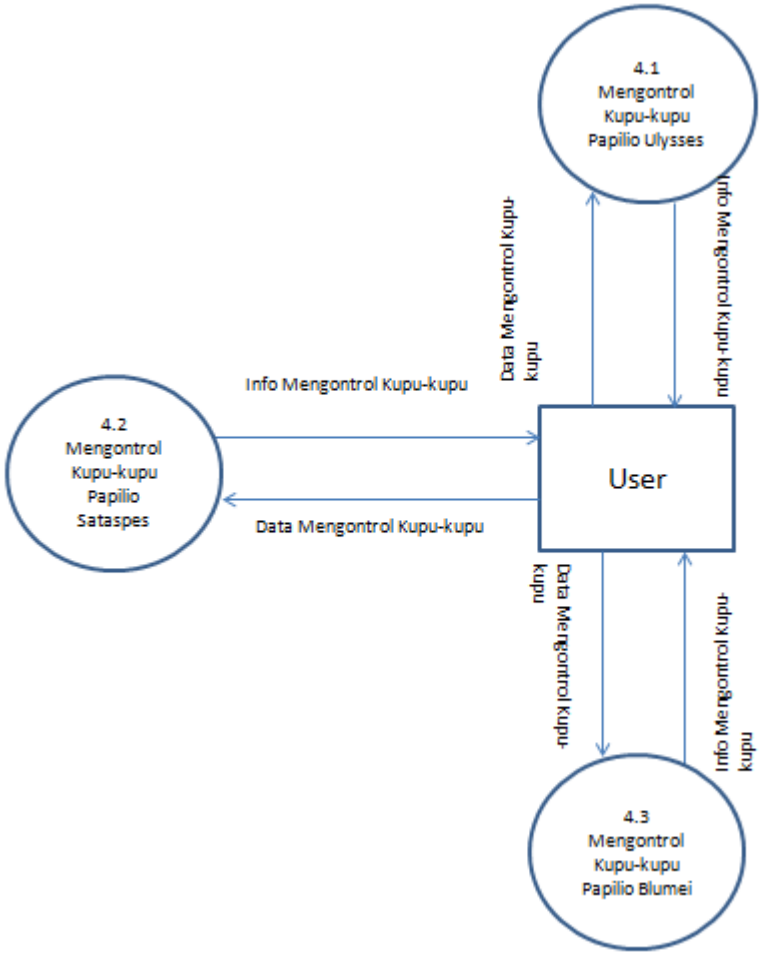
Gambar 3. 23 DFD Level 2 Proses 1.0 : Menampilkan Object Kupu-kupu

3.4.2 DFD Level 2 Proses 3.0 : Menampilkan Video Kupu-kupu



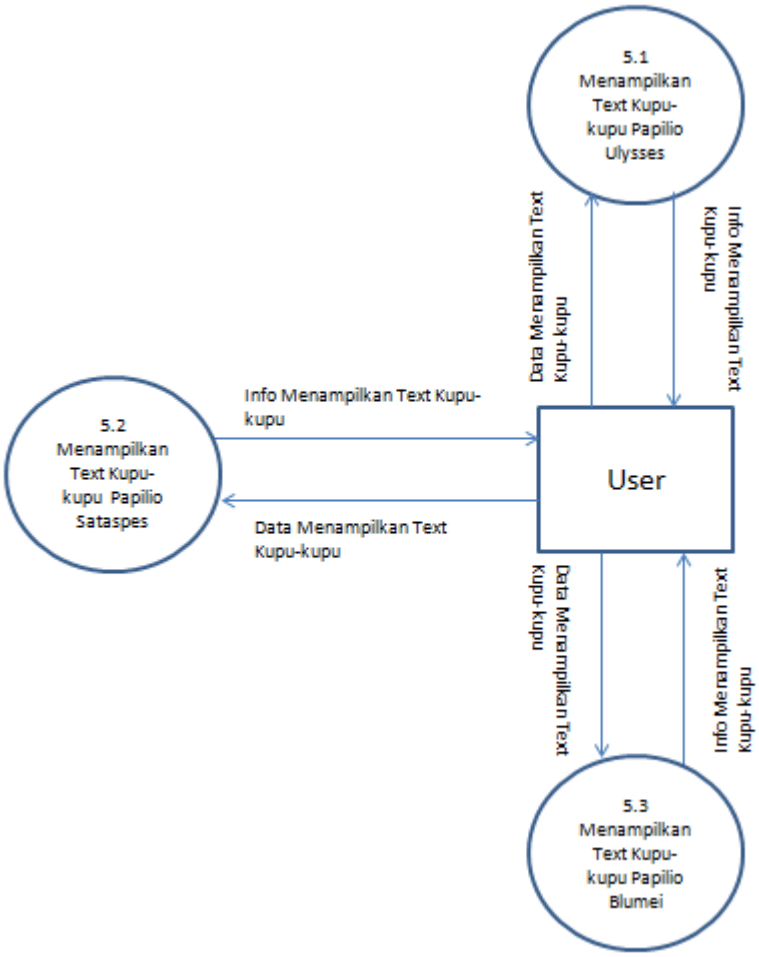
Gambar 3. 24 DFD Level 2 Proses 3.0 : Menampilkan Video Kupu-kupu

3.4.3 DFD Level 2 Proses 4.0 : Mengontrol Kupu-kupu



Gambar 3. 25 DFD Level 2 Proses 4.0 : Mengontrol Kupu-kupu

3.4.4 DFD Level 2 Proses 5.0 : Menampilkan Text Kupu-kupu



Gambar 3. 26 DFD Level 2 Proses 5.0 : Menampilkan Text Kupu-kupu

3.5 Spesifikasi Proses

Spesifikasi proses berfungsi untuk menjelaskan proses yang terdapat pada laboratorium virtual. Berikut spesifikasi proses dapat dilihat pada Tabel 3.2.

Tabel 3. 2 Tabel Spesifikasi Proses

No.	Proses	Keterangan
Proses Menampilkan Object Kupu-kupu		
1	No. Proses	1.0
	Nama Proses	Menampilkan Object Kupu-kupu
	Source	User
	Input	Data menampilkan Object Kupu-kupu
	Output	Info menampilkan Object Kupu-kupu
	Destination	User
	Logika Proses	1. Jika Kupu-kupu terinisialisasi pada database maka muncul object Kupu-kupu 2. Jika Kupu-kupu tidak terinisialisasi pada database maka tidak muncul object Kupu-kupu
Proses Verifikasi pola kupu-kupu		
2	No. Proses	2.0
	Nama Proses	Verifikasi pola kupu-kupu
	Source	Vuforia
	Input	Data pola Kupu-kupu
	Output	Info pola Kupu-kupu
	Destination	Vuforia
	Logika Proses	1. *User* mengarahkan kamera dari *handphone* ke koleksi kupu-kupu. 2. sistem melakukan proses *tracking* 3. Sistem mengecek kecocokan pola kupu-kupu yang berada di database dengan koleksi kupu – kupu yang di pindai kamera. 4. Jika Kupu-kupu terinisialisasi pada database maka akan menload data dari *dataset* 5. Jika Kupu-kupu tidak terinisialisasi pada database maka tidak akan melakukan perintah apa apa.
Proses Menampilkan Video Kupu-kupu		
	No. Proses	3.0
	Nama Proses	Menampilkan Video Kupu-kupu

3	Source	User
	Input	Data Menampilkan Video Kupu-kupu
	Output	Info Menampilkan Video Kupu-kupu
	Destination	User
	Logika Proses	1. Object Kupu-kupu terdeteksi oleh kamera pada *smartphone* 2. Sistem akan memunculkan Video dari setiap Kupu-kupu yang terdeteksi
Proses Mengontrol Kupu-kupu		
4	No. Proses	4.0
	Nama Proses	Mengontrol Kupu-kupu
	Source	User
	Input	Data Mengontrol Kupu-kupu
	Output	Info Mengontrol Kupu-kupu
	Destination	User
	Logika Proses	1. Object Terdeteksi oleh kamera *smartphone* 2. Sistem akan memunculkan button control yang memiliki fungsi untuk perbesar,perkecil,rotasi atas dan rotasi bawah dari setiap Kupu-kupu yang terdeteksi
Proses Menampilkan Text Kupu-kupu		
5	No. Proses	5.0
	Nama Proses	Menampilkan Text Kupu-kupu
	Source	User
	Input	Data Menampilkan Text Kupu-kupu
	Output	Info Menampilkan Text Kupu-kupu
	Destination	User
	Logika Proses	1. Object Kupu-kupu Terdeteksi oleh kamera *smartphone* 2. Sistem akan memunculkan button Text yang memiliki fungsi untuk memberikan informasi jenis Kupu-kupu dari setiap Kupu-kupu yang terdeteksi.
Proses Menampilkan Object Kupu-kupu Papilio Ulysses		

5	No. Proses	1.1
	Nama Proses	Menampilkan Object Kupu-kupu Papilio Ulysses
	Source	User
	Input	Data Menampilkan Object Kupu-kupu
	Output	Info Menampilkan Object Kupu-kupu
	Destination	User
	Logika Proses	1. Kupu-kupu telah terinisialisasi pada *database* 2. Sistem Mengecek object yang sudah di-apply di atas *image target.* 3. Jika Kupu-kupu Papilio Ulysses maka akan muncul object Kupu-kupu Papilio Ulysses
Proses Menampilkan Object Kupu-kupu Papilio Sataspes		
6	No. Proses	1.2
	Nama Proses	Menampilkan Object Kupu-kupu Papilio Sataspes
	Source	User
	Input	Data Menampilkan Object Kupu-kupu
	Output	Info Menampilkan Object Kupu-kupu
	Destination	User
	Logika Proses	1. Kupu-kupu telah terinisialisasi pada *database* 2. Sistem Mengecek object yang sudah di-apply di atas *image target.* 3. Jika Kupu-kupu Papilio Sataspes maka akan muncul object Kupu-kupu Papilio Sataspes
Proses Menampilkan Object Kupu-kupu Papilio Blumei		
7	No. Proses	1.3
	Nama Proses	Menampilkan Object Kupu-kupu Papilio Blumei
	Source	User
	Input	Data Menampilkan Object Kupu-kupu
	Output	Info Menampilkan Object Kupu-kupu
	Destination	User
	Logika Proses	1. Kupu-kupu telah terinisialisasi pada *database* 2. Sistem Mengecek object yang sudah di-

		apply di atas *image target.* 3. Jika Kupu-kupu Papilio Blumei maka akan muncul object Kupu-kupu Papilio Blumei.
Proses Menampilkan Video Kupu-kupu Papilio Ulysses		
8	No. Proses	3.1
	Nama Proses	Menampilkan Video Kupu-kupu Papilio Ulysses
	Source	User
	Input	Data Menampilkan Video Kupu-kupu
	Output	Info Menampilkan Video Kupu-kupu
	Destination	User
	Logika Proses	1. Kupu-kupu telah terinisialisasi pada *database* 2. Sistem Mengecek video yang sudah di-apply di atas *image target.* 3. Jika Kupu-kupu Papilio Ulysses maka akan muncul Video Kupu-kupu Papilio Ulysses.
Proses Menampilkan Video Kupu-kupu Papilio Sataspes		
9	No. Proses	3.2
	Nama Proses	Menampilkan Video Kupu-kupu Papilio Sataspes
	Source	User
	Input	Data Menampilkan Video Kupu-kupu
	Output	Info Menampilkan Video Kupu-kupu
	Destination	User
	Logika Proses	1. Kupu-kupu telah terinisialisasi pada *database* 2. Sistem Mengecek video yang sudah di-apply di atas *image target.* 3. Jika Kupu-kupu Papilio Sataspes maka akan muncul video Kupu-kupu Papilio Sataspes
Proses Menampilkan Video Kupu-kupu Papilio Blumei		
10	No. Proses	3.3
	Nama Proses	Menampilkan Video Kupu-kupu Papilio Blumei
	Source	User
	Input	Data Menampilkan Video Kupu-kupu
	Output	Info Menampilkan Video Kupu-kupu

	Destination	User
	Logika Proses	1. Kupu-kupu telah terinisialisasi pada *database* 2. Sistem Mengecek video yang sudah di-apply di atas *image target.* 3. Jika Kupu-kupu Papilio Blumei maka akan muncul video Kupu-kupu Papilio Blumei.
Proses Mengontrol Kupu-kupu Papilio Ulysses		
11	No. Proses	4.1
	Nama Proses	Mengontrol Kupu-kupu Papilio Ulysses
	Source	User
	Input	Data Mengontrol Kupu-kupu
	Output	Info Mengontrol Kupu-kupu
	Destination	User
	Logika Proses	1. Object Kupu-kupu Papilio Ulysses terdeteksi oleh kamera 2. *User* Mengontrol Object Kupu-kupu Papilio Ulysses 3. Jika menekan Tombol perbesar,maka Object kupu-kupu membesar ukurannya. 4. Jika menekan tombol perkecil,maka Object kupu-kupu memperkecil ukurannya. 5. Jika menekan tombol rotasi atas,maka Object kupu-kupu mengalami perubahan rotasi ke arah atas. 6. Jika menekan tombol rotasi bawah,maka Object kupu-kupu mengalami perubahan rotasi ke arah bawah.
Proses Mengontrol Kupu-kupu Papilio Sataspes		
12	No. Proses	4.2
	Nama Proses	Mengontrol Kupu-kupu Papilio Sataspes
	Source	User
	Input	Data Mengontrol Kupu-kupu
	Output	Info Mengontrol Kupu-kupu
	Destination	User
	Logika Proses	1. Object Kupu-kupu Papilio Sataspes terdeteksi oleh kamera 2. *User* Mengontrol Object Kupu-kupu

		Papilio Sataspes 3. Jika menekan Tombol perbesar,maka Object kupu-kupu membesar ukurannya. 4. Jika menekan tombol perkecil,maka Object kupu-kupu memperkecil ukurannya. 5. Jika menekan tombol rotasi atas,maka Object kupu-kupu mengalami perubahan rotasi ke arah atas. 6. Jika menekan tombol rotasi bawah,maka Object kupu-kupu mengalami perubahan rotasi ke arah bawah.
Proses Mengontrol Kupu-kupu Papilio Blumei		
13	No. Proses	4.3
	Nama Proses	Mengontrol Kupu-kupu Papilio Blumei
	Source	User
	Input	Data Mengontrol Kupu-kupu
	Output	Info Mengontrol Kupu-kupu
	Destination	User
	Logika Proses	1. Object Kupu-kupu Papilio Blumei terdeteksi oleh kamera 2. *User* Mengontrol Object Kupu-kupu Papilio Blumei 3. Jika menekan Tombol perbesar,maka Object kupu-kupu membesar ukurannya. 4. Jika menekan tombol perkecil,maka Object kupu-kupu memperkecil ukurannya. 5. Jika menekan tombol rotasi atas,maka Object kupu-kupu mengalami perubahan rotasi ke arah atas. 6. Jika menekan tombol rotasi bawah,maka Object kupu-kupu mengalami perubahan rotasi ke arah bawah.
Proses Menampilkan Text Kupu-kupu Papilio Ulysses		
14	No. Proses	5.1
	Nama Proses	Menampilkan Text Kupu-kupu Papilio Ulysses
	Source	User
	Input	Data Menampilkan Text Kupu-kupu
	Output	Info Menampilkan Text Kupu-kupu

	Destination	User
	Logika Proses	1. Object Kupu-kupu Papilio Ulysses terdeteksi setelah kamera di arahkan ke kupu-kupu Papilio Ulysses. 2. Sistem akan menampilkan tombol text pada layar. 3. Jika *user* menekan tombol text pada layar maka sistem menampilkan informasi mengenai Kupu-kupu Papilio Ulysses. 4. Jika *user* jika user tidak menekan tombol text pada layar.Maka sistem tidak akan menampilkan Informasi Kupu-kupu Papilio Ulysses.
Proses Menampilkan Text Kupu-kupu Papilio Sataspes		
15	No. Proses	5.2
	Nama Proses	Menampilkan Text Kupu-kupu Papilio Sataspes
	Source	User
	Input	Data Menampilkan Text Kupu-kupu
	Output	Info Menampilkan Text Kupu-kupu
	Destination	User
	Logika Proses	1. Object Kupu-kupu Papilio Sataspes terdeteksi setelah kamera di arahkan ke kupu-kupu Papilio Sataspes. 2. Sistem akan menampilkan tombol text pada layar. 3. Jika *user* menekan tombol text pada layar maka sistem menampilkan informasi mengenai Kupu-kupu Papilio Sataspes. 4. Jika *user* jika user tidak menekan tombol text pada layar.Maka sistem tidak akan menampilkan Informasi Kupu-kupu Papilio Sataspes.
Proses Menampilkan Text Kupu-kupu Papilio Blumei		
16	No. Proses	5.3
	Nama Proses	Menampilkan Text Kupu-kupu Papilio Blumei
	Source	User
	Input	Data Menampilkan Text Kupu-kupu

	Output	Info Menampilkan Text Kupu-kupu
	Destination	User
	Logika Proses	1. Object Kupu-kupu Papilio Blumei terdeteksi setelah kamera di arahkan ke kupu-kupu Papilio Blumei. 2. Sistem akan menampilkan tombol text pada layar. 3. Jika *user* menekan tombol text pada layar maka sistem menampilkan informasi mengenai Kupu-kupu Papilio Blumei. 4. Jika *user* jika user tidak menekan tombol text pada layar.Maka sistem tidak akan menampilkan Informasi Kupu-kupu Papilio Blumei.

3.6 Kamus Data

Kamus data merupakan sebuah daftar yang tersusun dari elemen data yang berhubungan dengan sistem. Kamus data dibuat berdasarkan arus data yang ada di data flow diagram. Berikut kamus data dapat dilihat pada tabel 3.3.

Tabel 3. 3 Tabel Kamus Data

No	Proses	Keterangan
1	Nama	Data Menampilkan Object Kupu-kupu
	Where used/how used	1. *User* – Proses Menampilkan Objek Kupu-kupu 1.0 2. *User* – Proses Menampilkan Objek Kupu-kupu Papilio Ulysses 1.1 3. *User* – Proses Menampilkan Objek Kupu-kupu Papilio Sataspes 1.2 4. *User* – Proses menampilkan Objek Kupu-kupu Papilio Blumei 1.3
	Deskripsi	Data yang digunakan ketika menampilkan objek Kupu-kupu Papilio Ulysses, Kupu-kupu Papilio Sataspes dan Kupu-kupu Papilio Blumei.
	Struktur data	- Koleksi Kupu-kupu
	Penjelasan struktur data	- Gambar[Kupu-kupu Papilio Ulysses | Kupu-kupu Papilio Sataspes |Kupu-kupu Papilio Blume]
2	Nama	Data Menampilkan Video Kupu-kupu
	Where used/how used	1. *User* – Proses Menampilkan Video

		Kupu-kupu 2.0 2. *User* – Proses Menampilkan Video Kupu-kupu Papilio Ulysses 2.1 3. *User* – Proses Menampilkan Video Kupu-kupu Papilio Sataspes 2.2 4. *User* – Proses menampilkan Video Kupu-kupu Papilio Blumei 2.3
	Deskripsi	Data yang digunakan ketika menampilkan video Kupu-kupu Papilio Ulysses, Kupu-kupu Papilio Sataspes dan Kupu-kupu Papilio Blumei.
	Struktur data	- Koleksi Kupu-kupu
	Penjelasan struktur data	- Gambar[Kupu-kupu Papilio Ulysses | Kupu-kupu Papilio Sataspes |Kupu-kupu Papilio Blume]
3	Nama	Data Mengontrol Kupu-kupu
	Where used/how used	1. *User* – Proses Mengontrol Kupu-kupu 3.0 2. *User* – Proses Mengontrol Kupu-kupu Papilio Ulysses 3.1 3. *User* – Proses Mengontrol Kupu-kupu Papilio Sataspes 3.2 4. *User* – Proses Mengontrol Kupu-kupu Papilio Blumei 3.3
	Deskripsi	Data yang digunakan ketika mengontrol Kupu-kupu Papilio Ulysses, Kupu-kupu Papilio Sataspes dan Kupu-kupu Papilio Blumei.
	Struktur data	- Koleksi Kupu-kupu
	Penjelasan struktur data	- Gambar[Kupu-kupu Papilio Ulysses | Kupu-kupu Papilio Sataspes |Kupu-kupu Papilio Blume]
4	Nama	Data Menampilkan Text Kupu-kupu
	Where used/how used	1. *User* – Proses Menampilkan Text Kupu-kupu 4.0 2. *User* – Proses Menampilkan Text Kupu-kupu Papilio Ulysses 4.1 3. *User* – Proses Menampilkan Text Kupu-kupu Papilio Sataspes 4.2 4. *User* – Proses Menampilkan Text Kupu-kupu Papilio Blumei 4.3
	Deskripsi	Data yang digunakan ketika Menampilkan Text Kupu-kupu Papilio Ulysses, Kupu-kupu Papilio

		Sataspes dan Kupu-kupu Papilio Blumei.
	Struktur data	- Koleksi Kupu-kupu
	Penjelasan struktur data	- Gambar[Kupu-kupu Papilio Ulysses | Kupu-kupu Papilio Sataspes |Kupu-kupu Papilio Blume]
5	Nama	Data pola Kupu-kupu
	Where used/how used	1. *Vuforia*– Proses Vefirikasi pola kupu-kupu
	Deskripsi	Data yang digunakan ketika pengecekan pola kupu-kupu
	Struktur data	- Koleksi Kupu-kupu
	Penjelasan struktur data	- Gambar[Kupu-kupu Papilio Ulysses | Kupu-kupu Papilio Sataspes |Kupu-kupu Papilio Blume]

3.7 Perancangan Sistem

Perancangan akan dimulai setelah tahap analisis terhadap sistem selesai dilakukan. Perancangan dapat didefinisikan sebagai penggambaran,perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi.

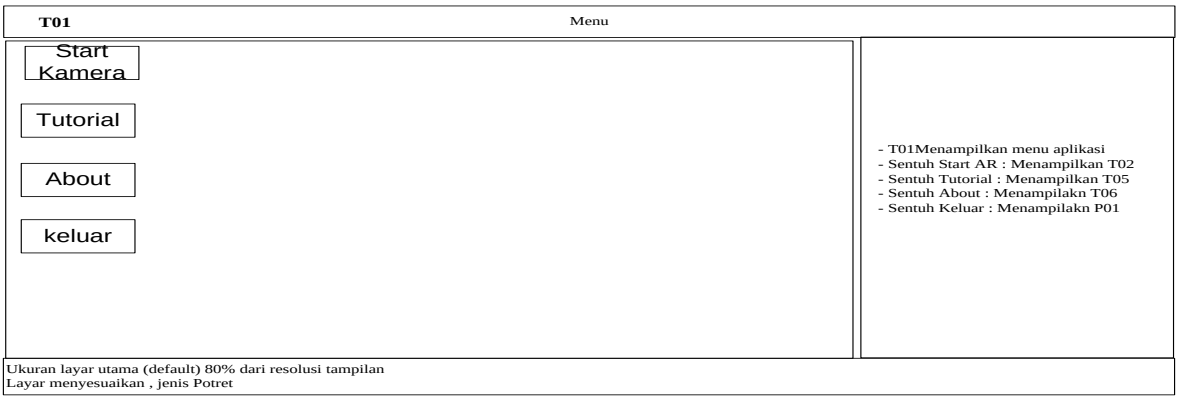
3.7.1 Perancangan Grafis dan Antarmuka

Antar muka merupakan tampilan dari suatu program aplikasi AR Kupu-kupu yang berperan sebagai media komunikasi yang digunakan sebagai sarana berdialog antara program dengan pengguna. Sistem yang akan dibangun diharapkan menyediakan *interface* yang mudah dipahami dan digunakan oleh pengguna.

Perancangan antarmuka mendeskripsikan rencana tampilan dari setiap form yang akan digunakan pada aplikasi AR Kupu-kupu. Perancangan antarmuka pada Kupu-kupu terdiri dari perancangan menu utama, perancangan start AR, perancangan Tutorial dan perancangan About.

3.7.1.1 Perancangan Antarmuka Menu Utama

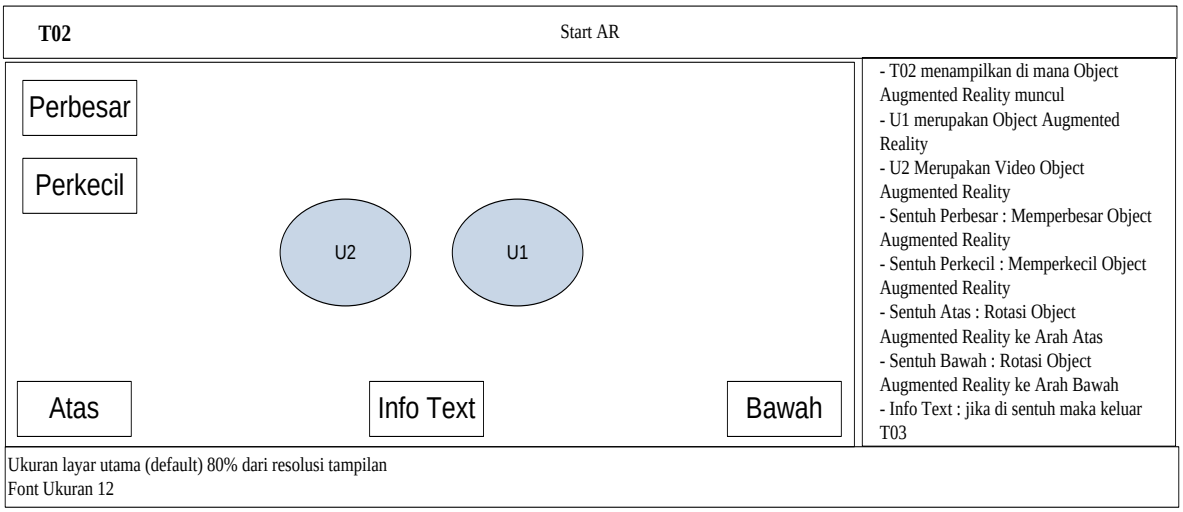
Perancangan menu utama mendeskripsikan rencana tampilan dari awal aplikasi *AR* Kupu-kupu dibuka. Berikut merupakan gambaran dari perancangan antarmuka aplikasi *AR* Kupu-kupu.



Gambar 3. 27 Perancangan Antar Muka Menu Utama

3.7.1.2 Perancangan Antarmuka Start Kamera

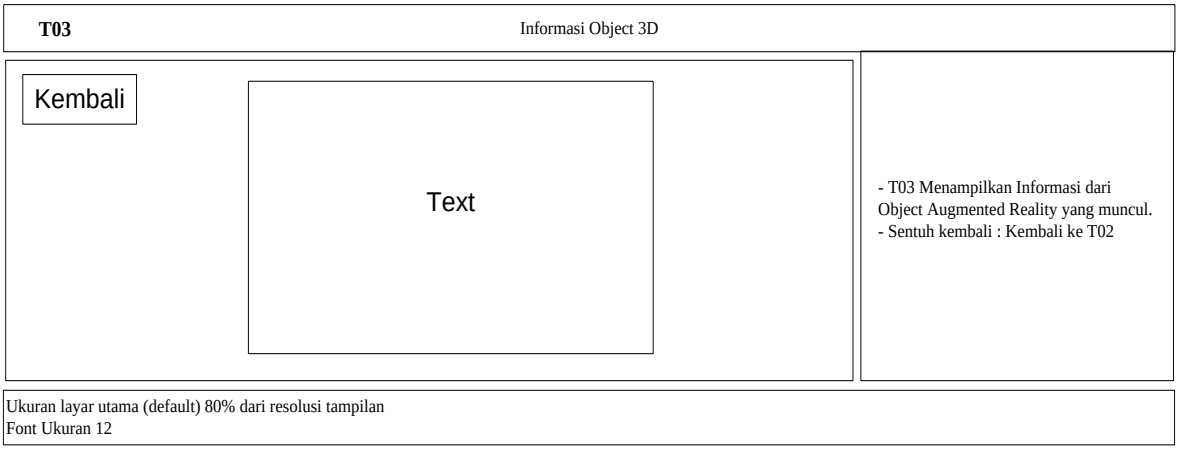
Perancangan start AR mendeskripsikan rencana tampilan dari permulaan ketika user mulai mengaktifkan kamera dan akan menscan Kupu-kupu .Berikut merupakan gambaran dari perancangan start AR aplikasi AR Kupu-kupu.



Gambar 3. 28 Perancangan Antar Muka Start Kamera

3.7.1.3 Perancangan Antarmuka Informasi Text

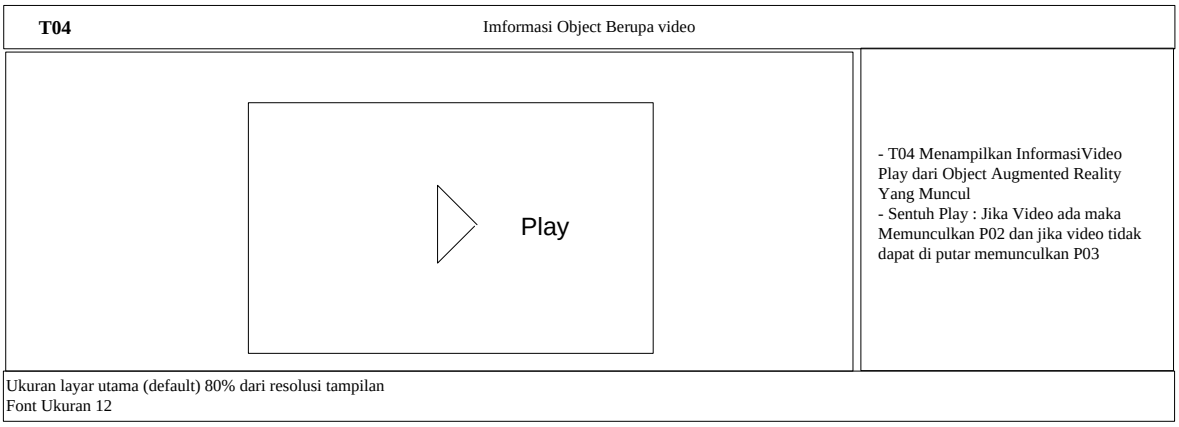
Perancangan informasi text mendeskripsikan rencana tampilan dari ketika objek 3D menapilkan informasi berupa text .Berikut merupakan gambaran dari perancangan antar muka informasi text pada Kupu-kupu.



Gambar 3. 29 Perancangan Antar Muka Informasi Text

3.7.1.4 Perancangan Antarmuka Play Video

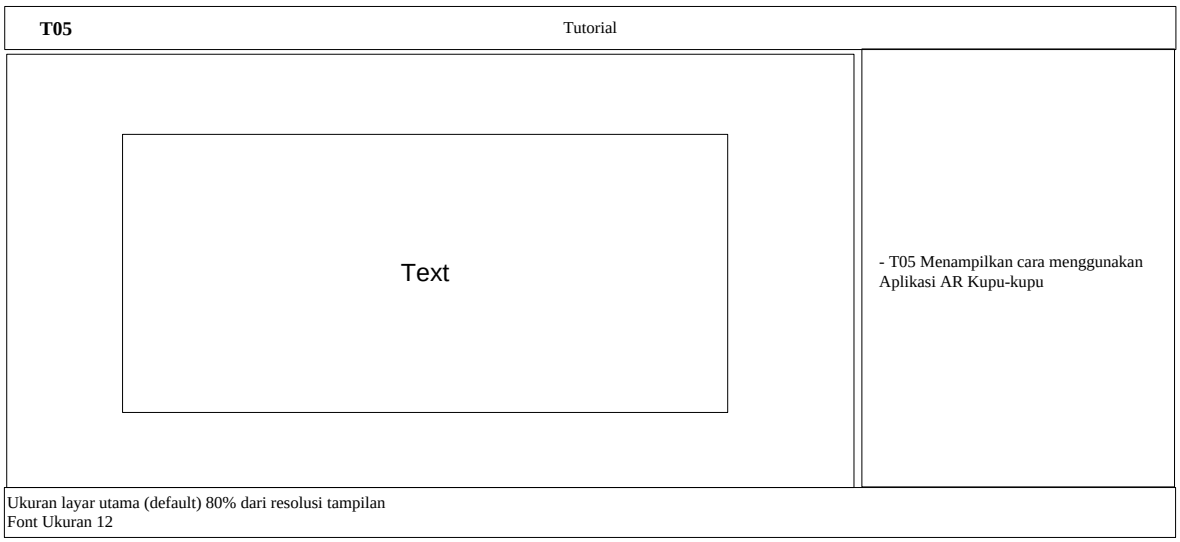
Perancangan play Video mendeskripsikan rencana tampilan dari ketika objek 3D menapilkan informasi berupa video .Berikut merupakan gambaran dari perancangan antar muka play video pada Kupu-kupu.



Gambar 3. 30 Perancangan Antar Muka Play Video

3.7.1.5 Perancangan Tutorial

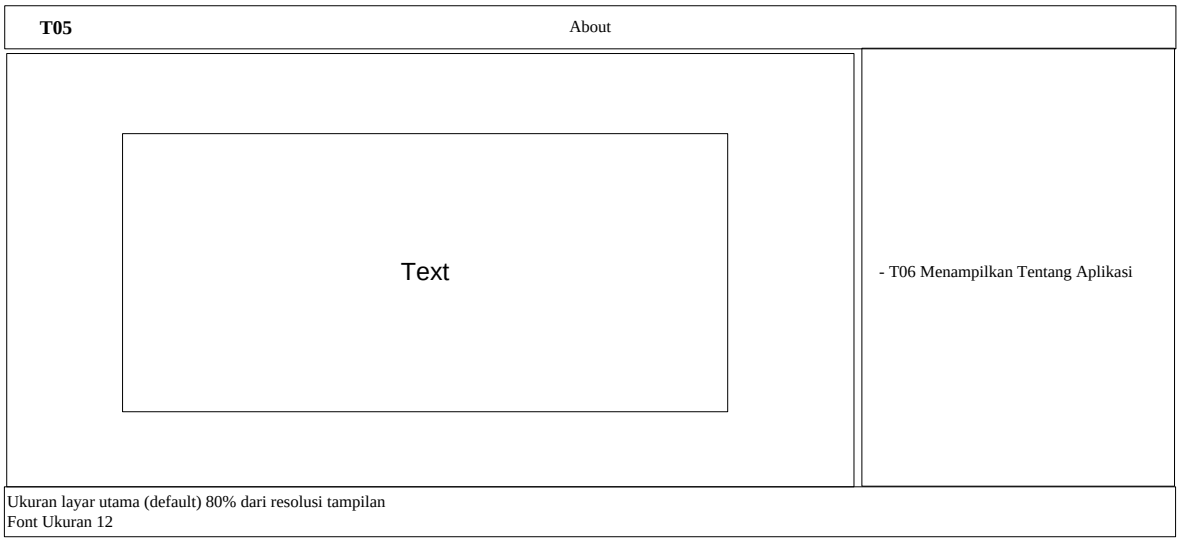
Perancangan antar muka Tutorial mendeskripsikan rencana tampilan dari petunjuk penggunaan aplikasi AR Kupu-kupu. Berikut merupakan gambaran dari perancangan petunjuk dari aplikasi AR Kupu-kupu.



Gambar 3. 31 Perancangan Antar Muka Tutorial

3.7.1.6 Perancangan About

Terakhir adalah Perancangan antar muka About mendeskripsikan rencana tampilan dari informasi aplikasi AR Kupu-kupu. Berikut merupakan gambaran dari perancangan antar muka about dari aplikasi AR Kupu-kupu.

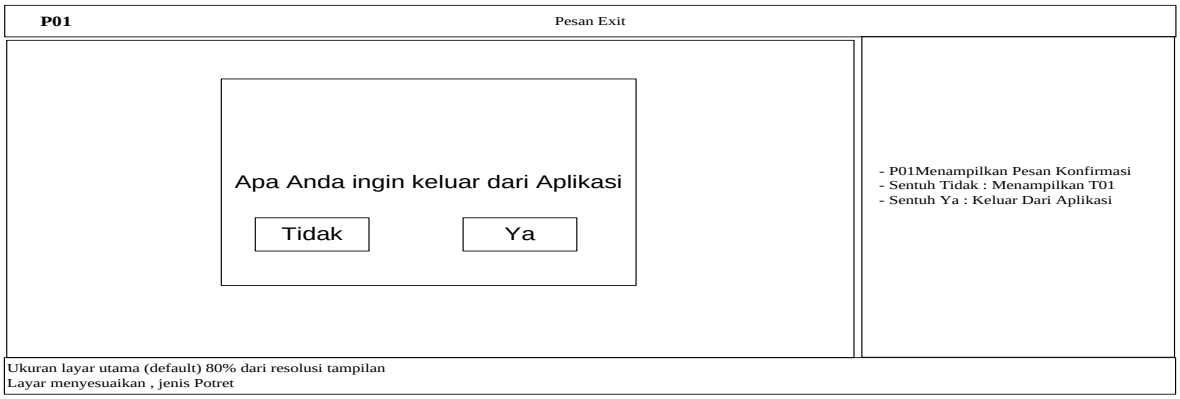


Gambar 3. 32 Perancangan Antar Muka About

3.7.1.7 Perancangan Pesan

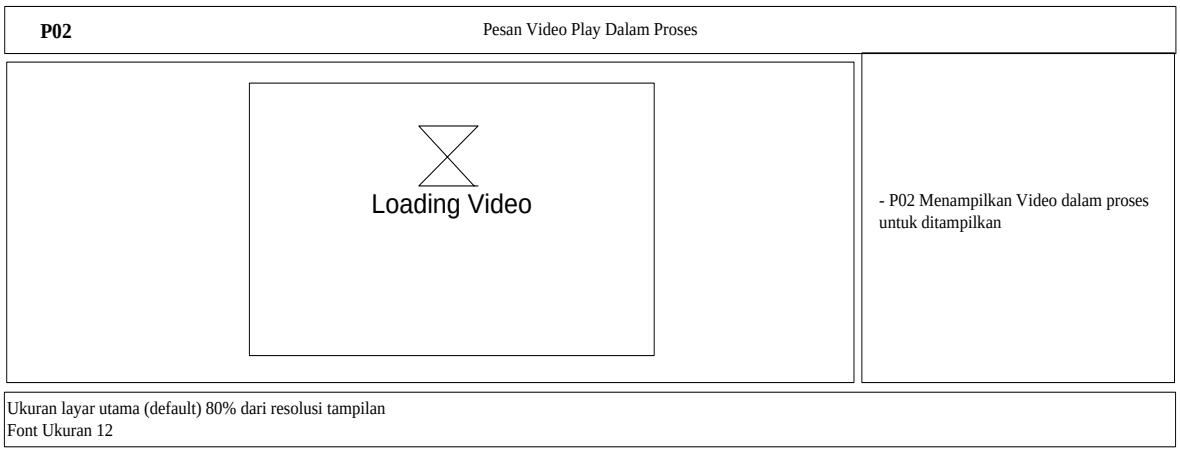
Pada gambar dibawah ini merupakan perancangan pesan yang terdapat pada aplikasi AR Kupu-kupu. Adapun tampilan pesan yang ada adalah sebagai berikut :

3.7.1.7.1 Perancangan Pesan Exit



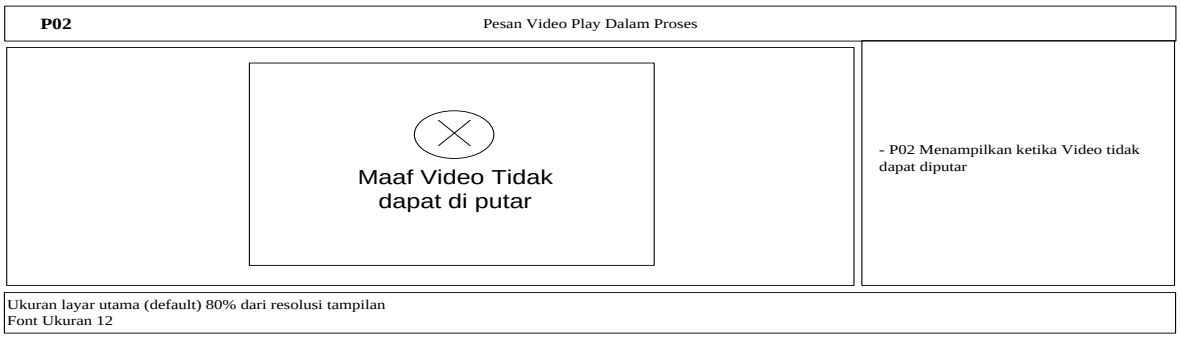
Gambar 3. 33 Perancangan Antarmuka Pesan Exit

3.7.1.7.2 Perancangan Pesan Video Play dalam Proses



Gambar 3. 34 Perancangan Video Play dalam Proses

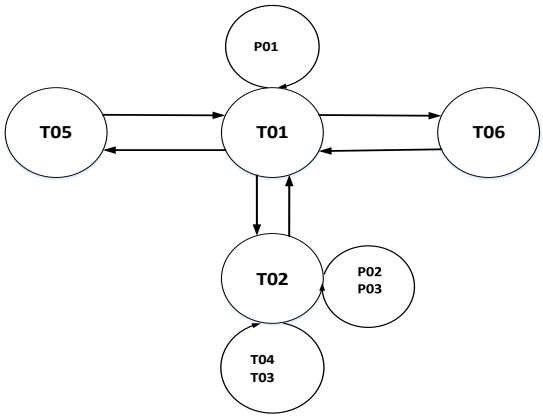
3.7.1.7.3 Perancangan Pesan Video Play Error



Gambar 3. 35 Perancangan Pesan Video Play Error

3.7.2 Jaringan Semantik

Jaringan semantik merupakan gambaran grafis yang menunjukkan antar objek (lingkaran) dan garis yang menggambarkan formasi antar objek tersebut. Adapun jaringan semantik untuk aplikasi AR Kupu-kupu dapat dilihat pada gambar dibawah:



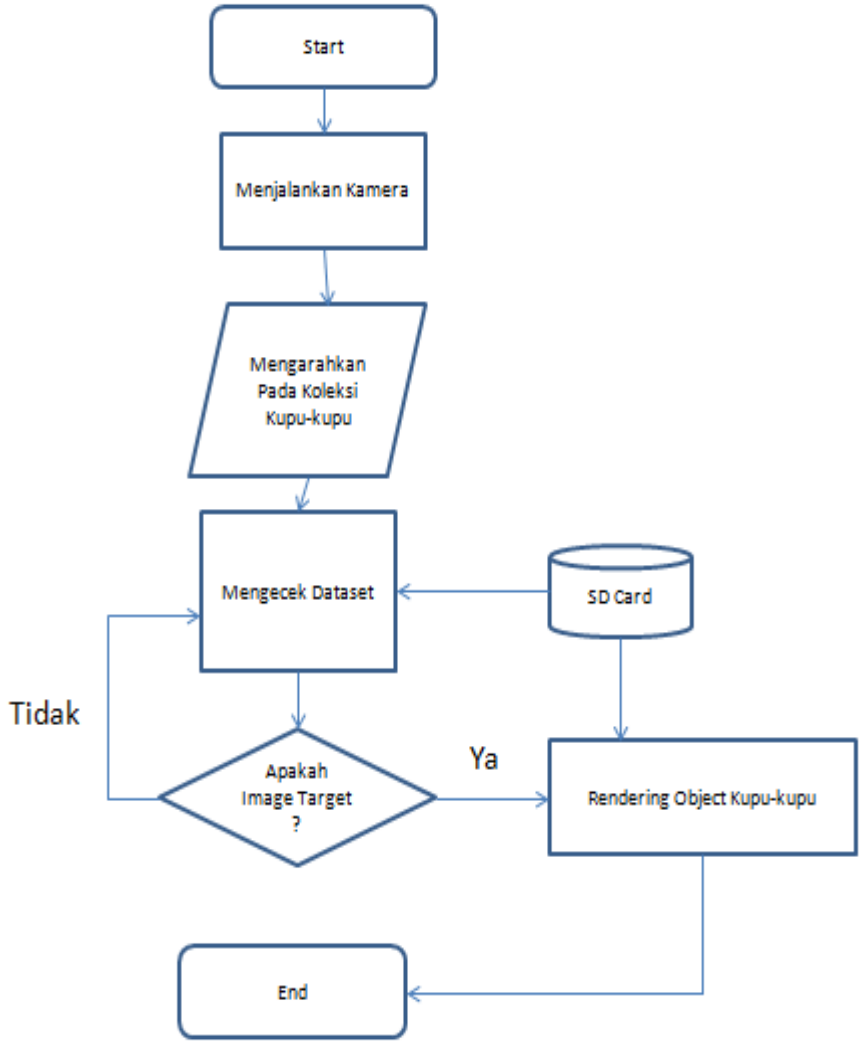
Gambar 3. 36 Jaringan semantik

3.7.3 Perancangan Prosedural

Pada bagian ini akan dibuat deskripsi rinci dari perangkat lunak, yaitu algoritma yang akan digunakan oleh masing-masing prosedur yang ada pada masing-masing modul. Perancangan prosedural menjelaskan tentang bagaimana program aplikasi berjalan. Adapun deskripsi rincinya adalah sebagai berikut :

3.7.3.1 *Flowchart* Menampilkan Object Kupu-kupu

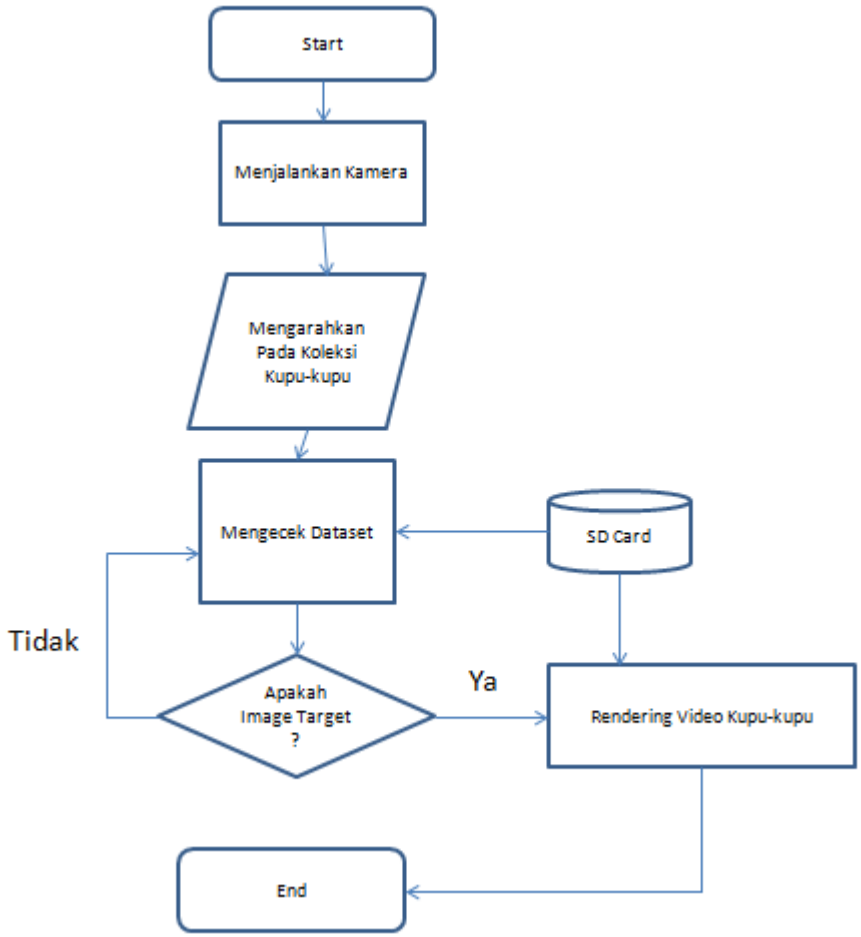
Flowchart pada gambar 3.37 menunjukkan proses dari sistem untuk menampilkan objek Kupu-kupu dari masing – masing Kupu-kupu.



Gambar 3. 37 *Flowchart* Menampilkan Object Kupu-kupu

3.7.3.2 *Flowchart* Menampilkan Video Kupu -kupu

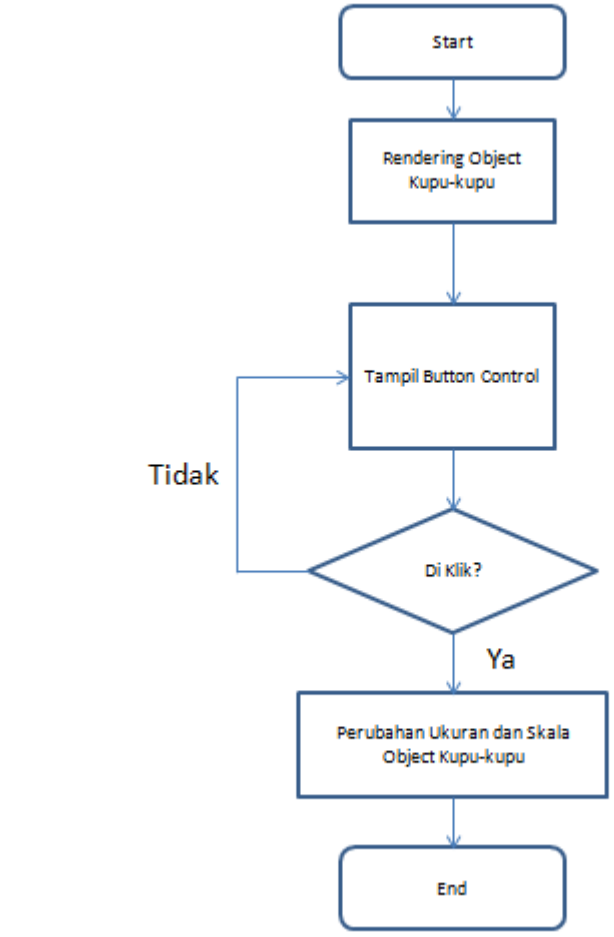
Flowchart pada gambar 3.38 menunjukkan proses dari sistem untuk menampilkan Video Kupu-kupu dari masing – masing Kupu-kupu.



Gambar 3. 38 *Flowchart* Menampilkan Video Kupu-kupu

3.7.3.3 *Flowchart* Mengontrol Kupu-kupu

Flowchart pada gambar 3.39 menunjukkan proses dari sistem untuk Mengontrol Object Kupu-kupu dari masing – masing Kupu-kupu.



Gambar 3. 39 *Flowchart* mengontrol Kupu-kupu