BAB II
LANDASAN TEORI

II.1.1 Batik

Menurut Kuswadji, batik berasal dari Jawa, "Mbatik", kata mbat dalam bahasa yang yang juga disebut ngembat. Arti kata tersebut melontarkan atau melemparkan. Sedangkan kata tik bisa diartikan titik. Jadi, yang dimaksud batik atau *mbatik* adalah melemparkan titik berkali-kali pada kain. Sedangkan menurut Soedjoko, batik berasal dari bahasa Sunda. Dalam bahasa Sunda, batik berarti menyunging pada kain dengan proses pencelupan. Istilah batik dalam bahasa sunda bisa ditemukan dalam *Babad Sengkala* (1633) dan *Pandji Djaja Lengkara* (1770) [5].

Batik mulai berkembang pada zaman kerajaan majapahit dan penyebaran islam di Jawa. Pada mulanya, batik hanya dibuat terbatas oleh kalangan kraton. Hasilnya kemudian dipakai oleh raja dan keluarga serta para pengikutnya. Kemudian batik dibawa keluar kraton oleh para pengikut raja maka dari itu kesenian batik mulai berkembang di masyarakat.

Pada abad ke-XIX setelah selesainya peperangan Dipenogoro, banyak pengikut-pengikut Dipenogoro yang meninggalkan daerah Yogyakarta dan berpindah menuju selatan. Sebagian dari mereka ada yang menetap di daerah Banyumas dan sebagian lainnya memutuskan menetap di daerah Ciamis dan Tasikmalaya. Pada saat itulah batik mulai menjadi bagian hidup masyarakat Ciamis dan Tasikmalaya, kerajinan membatik ditularkan kepada penduduk asli dan mulai berkembang di berbagai daerah sebagai komoditi kerajinan yang memiliki nilai jual tinggi [6].

II.1.2 Proses Membatik

Berikut merupakan proses-proses dalam membatik [1]:

1. Ngemplong Merupakan tahap pendahuluan. Diawali dengan mencuci kain mori untuk menghilangkan kanji, kemudian dilanjutkan dengan memasukkan kain mori ke dalam minyak jarak atau minyak kacang di dalam abu merang.

Proses ini disebut dengan *pengeloyoran*. Tujuan kain dimasukkan ke dalam minyak jarak agar karakter kain menjadi lemas sehingga lebih memiliki daya serap yang tinggi dalam menyerap zat warna. Setelah itu kain dijemur hingga kering, kemudian dilakukan proses *pengemplongan* yakni menghaluskan lapisan kain mori dengan cara dipukul menggunakan palu.

2. Nyorek (Memola) merupakan Proses membuat pola dengan cara meniru pola motif yang sudah ada. Proses ini sering disebut dengan *ngeblat*. Pola biasa dibuat di atas kertas roti sebelum dijiplak di atas kain mori. Untuk menjiplak menggunakan alat pensil atau canting. Agar proses pewarnaan sempurna maka pola harus dibuat di kedua sisi kain. Proses ini disebut dengan *ganggang*.
3. Mbathik merupakan tahap menorehkan malam ke kain mori. Dimulai dengan menggambar garis-garis di luar pola (*nglowong*) dan mengisi pola dengan bermacam bentuk (isen-isen).
4. Nembok merupakan proses menutupi bagian-bagian yang tidak boleh terkena warna dasar, dengan menggunakan lapisan malam yabg tebal.
5. Medel merupakan proses pencelupan kain yang sudah dibatik ke cairan warna secara berulang-ulang untuk mendapatkan warna yang diinginkan.
6. Ngerok dan Mbirah merupakan proses malam pada kain dikerok secara hati-hati menggunakan lempengan logam, kemudian dibilas dengan air bersih. Setelah itu kain diangin-anginkan.
7. Mbironi merupakan proses menutupi warna biru dan isen-isen. Selain itu ada proses *ngrining* yakni mengisi bagian yang belum diwarnai dengan motif tertentu. Proses ngrining biasa dilakukan setelah proses pewarnaan.
8. Menyoga merupakan proses untuk mendapatkan warna coklat dengan cara mencelupkan kain ke dalam campuran warna coklat. Menyoga berasal dari kata soga, sejenis kayu untuk mendapatkan warna cokelat.
9. Nglorod merupakan tahap akhir dalam pembuatan kain batik. Tahap ini bertujuan melepaskan seluruh malam (lilin) dengan cara memasukkan kain yang sudah cukup tua warnanya ke dalam air mendidih.Setelah diangkat, kain dibilas dengan air bersih, kemudian diangin-anginkan hingga kering.

II.1.3 Macam- Macam Batik Jawa Barat

Jawa Barat merupakan daerah yang terkenal dengan keragamaan batiknya. Berikut adalah macam-macam motif batik khas Jawa Barat:

1. Motif Batik Cirebon

Motif batik Megamendung merupakan karya seni batik yang identik dan bahkan menjadi ikon batik daerah Cirebon. Motif megamendung yang pada awalnya selalu berunsurkan warna biru diselingi warna merah menggambarkan maskulinitas dan suasana dinamis, karena dalam proses pembuatannya ada campur tangan laki-laki. Kaum laki-laki anggota tarekatlah yang pada awalnya merintis tradisi batik. Warna biru dan merah tua juga menggambarkan psikologi masyarakat pesisir yang lugas, terbuka dan egaliter, Gambar II.1 merupakan contoh batik khas cirebon [1].



Gambar II.1 Motif Batik Mega Mendung Cirebon

2. Motif Batik Bogor

Motif-motif batik Bogor terinspirasi dari peninggalan kerajaan Pakuan, benda-benda sejarah, fenomena alam, dan kebudayaan. Salah satu motif yang terkenal adalah motif Kujang Kijang. Motif ini mengandung dua ikon kota Bogor, yaitu Kujang dan Kijang. Kujang merupakan senjata tradisional khas Sunda, sedangkan kijang merupakan hewan yang berada di Istana Bogor [7]. Berikut adalah salah satu gambar motif kujang kijang merah biru yang terdapat pada Gambar II.2.



Gambar II.2 Motif Batik Kijang Kujang Bogor

3. Motif Batik Ciamis

Batik ciamisan tampil sederhana tapi penuh wibawa. Kesederhaan ini tak lepas dari sejarah keberadaannya yang banyak dipengaruhi daerah lain, seperti ragam hias pesisiran dari Indramayu dan Cirebon. Salah satu motif batik khas Ciamis adalah motif batik parang rusak. Motif parang rusak merupakan motif batik yang diciptakan oleh Panembahan Senopati pada saat bertapa di laut selatan. Motif tersebut terinspirasi dari ombak yang tidak pernah lelah menghantam karang [7]. Berikut adalah gambar motif batik parang rusak yang terdapat pada Gambar II.3.



Gambar II.3 Motif Batik Parang Rusak Ciamis

4. Motif Batik Garut

Motif batik garutan umumnya menghadirkan ragam hias datar bentuk- bentuk geometrik. Bentuk-bentuk geometrik ini mengarah secara diagonal, bentuk kawung, atau belah ketupat. Ada pula motif-motif yang mengambil pola bentuk-bentuk flora dan fauna.Bentuk motif batik Garut merupakan cerminan dari kehidupan sosial budaya, falsafah hidup, dan adat-istiadat orang Sunda [7]. Beberapa perwujudan batik Garut secara visual dapat digambarkan melalui motif

dan warnanya. Salah satunya adalah motif batik rereng ageung yang teradapat pada Gambar II.3



Gambar II.4 Motif Batik Rereng Ageung

5. Motif Batik Tasikmalaya

Motif batik Tasikmalaya sangat kental dengan nuansa Parahyangan, seperti bunga anggrek dan burung. Selain itu ada juga motif Merak-Ngibing, Cala-Culu, Pisang-Bali, Sapujagat, dan Awi Ngarambat. Batik Tasik memiliki kekhususan tersendiri, yaitu bermotif alam, flora, dan fauna. Batik Tasik hampir sama dengan Batik Garut, hanya berbeda dari sisi warnanya yang lebih terang. Motif Merak Ngibing sendiri mengandung makna keindahan dan kepatutan. Namun ada juga yang menyebutkan Merak Ngibing sebagai ekspresi kebahagian dan luapan kegembiraan masyarakat sunda yang divisualisasikan dalam gambar/motif merak yang sedang menari [7] . Berikut adalah gambar motif batik merak ngibing yang terdapat pada gambar Gambar II.5.

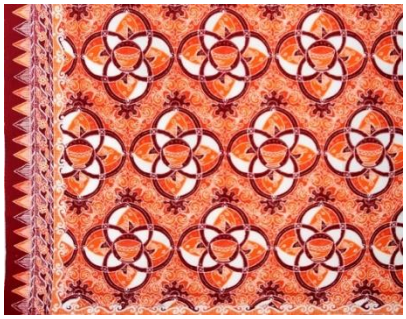


Gambar II.5 Motif Batik Merak Ngibing Tasikmalaya

6. Motif Batik Kuningan

Pada akhir tahun 2011, beberapa motif batik Kuningan diresmikan dari sebuah lomba cipta desain batik Kuningan. Salah satu dari motif-motif-motif baru tersebut adalah motif ikan dewa. Kini disepakati sebagai motif batik khas daerah

Kuningan sehingga bisa dikembangkan dan dijadikan pakaian resmi bagi kalangan PNS maupun pegawai swasta bahkan para siswa sekolah yang ada di Kab. Kuningan [7]. Gambar motif batik ciamis terdapat pada gambar II.7.



Gambar II.6 Motif Batik Ikan Dewa Kuningan [1]

II.2 Game Edukasi

Game edukasi merupakan suatu kegiatan yang sangat menyenangkan dan dapat merupakan cara atau alat pendidikan yang bersifat mendidik [8]. Tujuan dari *game* edukasi adalah memancing minat belajar pelajar terhadap materi pelajaran sambil bermain sehingga dengan perasaan senang diharapakan pelajar dapat lebih mudah memahami materi pelajaran yang disajikan. Berdasarkan pengertian tersebut, karakteristik dari *game* edukasi itu sendiri adalah menyenangkan, memotivasi, membuat kecanduan dan kolaboratif membuat aktifitas tersebut digemari oleh banyak orang. Saat ini perkembangan *game* sudah banyak menyentuh banyak bidang, *game* dianggap sebagai media interaktif yang efektif untuk menyampaikan pesan.

Ada beberapa prinsip yang harus diterapkan dalam aplikasi sebuah *game* edukasi sebagai berikut [9]:

1. *Individualization*

Materi pembelajaran (pengetahuan) dibuat sesuai dengan kebutuhan individual dari pembelajar, sedangkan *game* mengadopsi level individual dari Pengguna.

2. *Feedback Active*

Adanya *feedback* yang sesuai dengan cepat untuk memperbaiki pembelajaran dan mengurangi ketidaktahuan pelajar terhadap materi yang disampaikan, sedangkan *game* menyediakan dengan cepat dan kontes.

3. *Active learning*

Adanya kecendrungan untuk menyertakan pelajar secara active dalam menciptakan penemuan dan pengetahuan baru yang membangun, sedangkan *game* menyediakan suatu lingkungan yang membantu terjadinya penemuan.

4. *Motivation*

Pelajar termotivasi dengan *reward* yang diberikan dalam aktivitas permainan, sedangkan *game* melibatkan pengguna berjam-jam untuk mencapai tujuan.

5. *Social*

Pengetahuan merupakan suatu proses partisipasi sosial, sedangkan *game* dapat dimainkan dengan orang lain (seperti *game multiplayer*) atau melibatkan komunitas dari pecinta *game* yang sama.

6. *Scaffolding*

Pelajar secara berangsur-angsur diarahkan dengan tingkat kesulitan yang makin tinggi dan dapat melangkah lebih maju untuk mencapai kemenangan dari permainan, sedangkan game dibangun secara multilevel, Pengguna tidak bisa bergerak ke level yang lebih tinggi sampai dia mampu menyelesaikan permainan di level yang ada.

7. *Transfer*

Pelajar mengembangkan kemampuan untuk mentransfer pengetahuan dari satu orang ke orang yang lain, sedangkan game mengijinkan Pengguna untuk mentransfer informasi dari suatu konteks ke konteks yang lain.

II.3 Aplikasi Mobile

Aplikasi *mobile* adalah sebuah aplikasi yang memungkinkan pengguna melakukan mobilitas dengan mudah dari satu tempat ke tempat lain tanpa terjadi pemutusan atau terputusnya komunikasi. Aplikasi ini dapat diakses dengan

menggunakan perlengkapan seperti PDA, telepon seluler atau *handphone*. Sedangkan menurut Bentley pengertian *aplikasi mobile* adalah:

> "Aplikasi mobile adalah sebuah bahasa pemrograman yang mempresentasikan apa yang seharusnya dilakukan oleh perangkat lunak atau bagaimana suatu proses perangkat lunak seharusnya menyelesaikan tugasnya [10]".

Adapun karakteristik dari aplikasi mobile adalah sebagai berikut [10]:

1. Konektivitas: Aplikasi *mobile* tergantung terhadap konektivitas lokasi *provider*, maka disarakan aplikasi mobile yang baik adalah yang mengambil *bandwith* rendah atau bahkan mungkin tidak perlu menggunakan *bandwith*.
2. Kenyamanan: Aplikasi *mobile* yang baik dapat melakukan tugasnya dengan cepat pada berbagai situasi walaupun pada konteks berbeda-beda seperti pengaturan cahaya dan kebisingan, respon terhadap jari pengguna dan goyah terhadap gerakan perangkat.
3. Ketersediaan: Aplikasi *mobil*e yang baik dapat digunakan dimana saja dan kapan saja dengan ketersediaan informasi yang sesuai dengan konten.
4. Personalisasi: Menciptakan konten pada aplikasi *mobile* lebih dipersonalisasikan pada karakterstik pengguna awal (*native user*). Pembuatan fitur lebih dilatarbelakangi oleh kategori pengguna, kebutuhan ini bukan hanya meliputi materi yang bersifat pribadi namun juga meliputi kontrol data.

II.4 Mobile Learning

Mobile learning (m-learning) adalah pembelajaran yang memanfaatkan teknologi dan perangkat *mobile*. Dalam hal ini, perangkat *mobile* dapat berupa PDA, handphone, laptop, tablet PC, dsb. Dengan *mobile learning*, pengguna dapat mengakses konten pembelajaran dimana saja dan kapan saja, tanpa harus mengunjungi suatu tempat pada waktu terntu. Menurut Clark Quinn *mobile learning* adalah:

> "*The intersection of mobile computing and e-learning: accessible resources wherever you are, strong search capabilities, rich interaction, powerful support for effective learning, and performance -based assessment. E-Learning independent of location in time or space* [11]*".*

Pada dasarnya konsep pembelajaran *mobile learning* adalah membawa manfaat ketersediaan materi ajar yag dapat diakses setiap saat dan visualisasi yang menarik. Adapun kemampuan penting yang harus disediakan adalah kemampuan menyajikan informasi pembelajaran dan kemampuan merealisasikan komunikasi bilateral anatara pengajar dan pelajar. Tujuan dari mobile learning itu sendiri adalah proses belajar sepanjang waktu (*long life learning*) sehingga dalam penggunaannya pelajar dapat lebih aktif belajar dan lebih menghemat waktu karena tidak perlu seorang pelajar hadir dalam suatu ruang belajar.

II.5 Kuesioner

Angket atau kuesioner adalah instrumen penelitian yang berupa daftar pertanyaan untuk memperoleh keterangan dari sejumlah responden (sumber yang diambil datanya melalui angket). Angket atau kuesioner dapat disebut sebagai wawancara tertulis, karena isi kuesioner merupakan satu rangkaian pertanyaan tertulis yang ditujukan kepada responden dan diisi sendiri oleh responden [12].

Pada penilaian hasil kuesioner ditetapkan bentuk pengukuran skala kuesioner. Penskalaan adalah proses menetapkan nomor-nomor atau simbol-simbol terhadap suatu atribut atau karakteristik yang bertujuan untuk mengukur atribut atau karakteristik tersebut. Alasan penganalisis sistem mendesain skala adalah untuk mengukur sikap atau karakteristik orang-orang yang menjawab kuesioner dan untuk respoden memilih subjek kuesioner.

Berikut adalah bentuk skala pengukuran dalam kuesioner yaitu :

1. Skala Likert

Skala likert adalah skala yang digunakan untuk mengukur sikap, pendapat, dan presepsi seseorang atau sekelompok orang tentang suatu gejala atau fenomena pendidikan. Ada dua bentuk pertanyaan yang menggunakan skala likert yaitu bentuk pertanyaan untuk mengukur sikap positif, dan bentuk pertanyaan negatif untuk mengukur sikap negatif. Berikut adalah tabel contoh pertanyaan skala likert [12].

Tabel II.1 Contoh Bentuk Pertanyaan Skala Likert

Pertanyaan Positif		Pertanyaan Negatif	
Sangat Setuju (SS)	5	Sangat Setuju (SS)	1
Setuju (S)	4	Setuju (S)	2
Netral (N)	3	Netral (N)	3
Tidak Setuju	2	Tidak Setuju	4
Sangat Tidak Setuju	1	Sangat Tidak Setuju	5

2. Skala Guttman

Skala guttman adalah skala yang digunakan untuk pengukuran yang menginginkan tipe jawaban tegas seperti jawaban benar-salah, pernah–tidak pernah, Selain dapat dibuat dalam pertanyaan pilihan ganda, skala guttman dapat juga dibuat dalam daftar checklist. Penilaian pertanyaan positif bagi pilihan benar akan mendapat nilai 1 dan yang salah akan mendapat nilai 0. Sedangkan untuk pertanyaan negatif, pilihan salah akan mendapat nilai 1 dan pilihan benar akan mendapatkan nilai 0 [12].

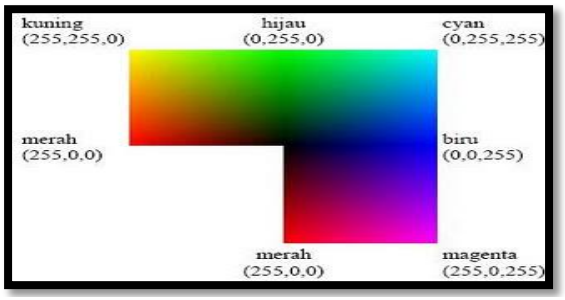
II.6 Citra Digital RGB

Citra digital dapat didefinisikan sebagai fungsi dua variabel f(x,y) dimana x dan y adalah koordinat spasial dan nilai f(x,y) merupakan intensitas citra pada koordinat tersebut. RGB adalah suatu model warna yang terdiri dari merah, hijau, dan biru. Setiap piksel disusun oleh tiga komponen warna R (red), G (Green), B (Blue). Kombinaasi dari tiga warna RGB tersebut menghasilkan warna yang khas untuk piksel yang bersangkutan. Pada citra 256 warna, setiap piksel panjangnya 8 bit, tetapi komponen warna RGB-nya disimpan di dalam tabel RGB yang disebut palet [13].

Di dalam citra RGB Setiap warna dasar, misalnya merah, dapat diberi rentang-nilai. Untuk monitor komputer, nilai rentangnya paling kecil = 0 dan paling besar = 255. Pilihan skala 256 ini didasarkan pada cara mengungkap 8 digit bilangan biner yang digunakan oleh mesin komputer. Dengan cara ini, akan diperoleh warna campuran sebanyak 256 x 256 x 256 = 1677726 jenis warna. Sebuah jenis warna, dapat dibayangkan sebagai sebuah vektor di ruang 3 dimensi yang biasanya dipakai dalam matematika, koordinatnya dinyatakan dalam bentuk tiga bilangan, yaitu komponen-x, komponen-y dan komponen-z. Misalkan sebuah

vektor dituliskan sebagai r = (x,y,z). Untuk warna, komponen-komponen tersebut digantikan oleh komponen R(ed), G(reen), B(lue). Jadi, sebuah jenis warna dapat dituliskan sebagai berikut:

warna = RGB(30, 75, 255). Putih = RGB (255,255,255), sedangkan untuk hitam= RGB(0,0,0).Bentuk Representasi warna dari sebuah citra digital pada Gambar II.7



Gambar II.7 Representasi Warna RGB Citra Digital [13]

.

II.7 Metode Sobel

Metode sobel adalah salah satu metode deteksi tepi hasil pengembangan metode deteksi tepi – robbert dengan menggunakan filter HPF *(High Pass Filter)* yang diberi satu angka nol penyangga [14]. Metode ini mengambil prinsip dari fungsi laplacian dan gaussian yang dikenal sebagai fungsi untuk membangkitkan HPF. Kelebihan Dari operator ini adalah kemampuan untuk mengurangi noise sebelum melakukan perhitungan deteksi tepi. Proses yang dilakukan dalam mendeteksi tepi dimulai dari masukan gambar asli kemudian akan dirubah menjadi citra grayscale, sobel, dilasi, closing, invers, erosi dan rekontruksi.

Deteksi tepi yang dilakukan operator sobel menggunakan dua *maks*, vertikal dan horizontal dengan matriks 3x3 [14].

$$Sy = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad Sx = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Operator sobel menempatkan penekanan atau pembobotan pada piksel -piksel yang lebih dekat dengan titik pusat jendela. Dengan demikian pengaruh piksel-piksel tetangga akan berbeda sesuai dengan letaknya terhadap titik di mana gradien dihitung. Gradien adalah hasil pengukuran perubahan dalam sebuah fungsi intensitas, dan sebuah citra dapat dipandang sebagai kumpulan beberapa fungsi

intensitas kontinyu dari citra. Dari susunan nilai-nilai pembobotan pada jendela juga terlihat bahwa perhitungan terhadap gradien juga merupakan gabungan dari posisi horisontal dan vertikal.

II.8 Windows Phone

Pada tanggal 15 Februari 2010 di pameran MWC, Barcelona, Microsoft memperkenalan sebuah sistem operasi yang dianggap sebagai *a revolutionary new platform* yaitu *Windows phone*. Berdasarkan hasil kerja sama antara CEO Microsoft Steve Ballmer dan CEO Nokia Stephen Elop, *Windows phone* merupakan sebuah sistem operasi *mobile* yang diterapkan pada ponsel genggam merk Nokia.

Pada *Windows Phone* ini Microsoft menghadirkan antarmuka pengguna yang tampak lebih *clean* dan *fresh*. Dengan filosofi desain yang dinamakan metro, terinsiprasi dari tanda-tanda yang terdapat pada metro *subway*, antarmuka *Windows phone* menunjukan ciri yang jelas, informasi yang mudah diperoleh, intuitif dan menggunakan simbol-simbol yang mudah dipahami. Disisi platform pengembangan sendiri *windows phone* memiliki spesifikasi yang cukup mumpuni seperti layar sentuh, sensor GPS, *accelerometer*, kompas, cahaya, kamera, multimedia, GPU dengan directX9, dan tiga *hardware button* [15].

II.9 Sensor Touch

Perangkat yang menjalankan *Windows phone* diantaranya wajib memiliki layar sentuh (*touch screen*) yang memungkinkan pengguna menggunakan beberapa jari secara bersama-sama untuk menghasilkan gesture yang berbeda seperti *tapping, flicking* atau *pinching*. Pergerakan jari akan mempengaruhi pola batik yang akan digambar oleh pengguna [15]. Berikut adalah Tabel II.2 merupakan gerakan yang bisa dilakukan dengan *touch* sebagai berikut:

Tabel II.2 Jenis *gesture sensor touch* [15]

Gerakan (Gesture)	Deskripsi
Tap	Gerakan sebuah jari menyentuh layar dan segera melepaskannya
Double Tap	Gerakan dua tap secara berurutan
Hold	Gerakan sebuah jari menyentuh dab menahan pada suatu tempat dalam periode waktu tertentu
Flick	Sebuah gerakan menyentuh dan menggerakan serta mengangkat jari
Pinch	Dua jari menekan layar dan menggerakannya

II.10 Structure Analysis and Design Technique

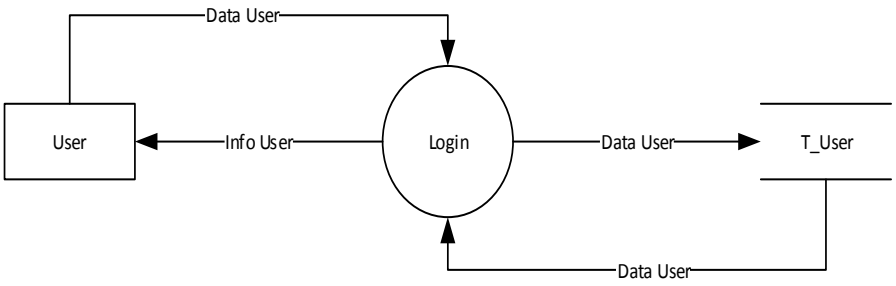
SADT adalah suatu metodologi yang dikembangkan oleh D. T. Ross pada tahun 1969 – 1973. SADT memandang suatu sistem terdiri dari dua hal yaitu objek dan kejadian. Yang termasuk kedalam objek suatu sistem informasi adalah data atau dokumen, dan yang termasuk kejadian adalah kegiatan yang dilakukan manusia, mesin, atau software.Untuk menggambarkan dua hal tersebut digunakan diagram yang berbeda, yaitu Diagram Data dan Diagram Kegiatan.

SADT sebagai metode terstruktur juga mengatur konsep Dekomposisi, dengan pendekatan top-down, yaitu melalui penggambaran sistem secara utuh (whole system) sebagai tingkat tertinggi (top level) dan kemudian memecahnya secara rinci [16].

Beberapa alat yang dapat digunakan pada metodologi pengembangan sistem yang terstruktur [16]adalah:

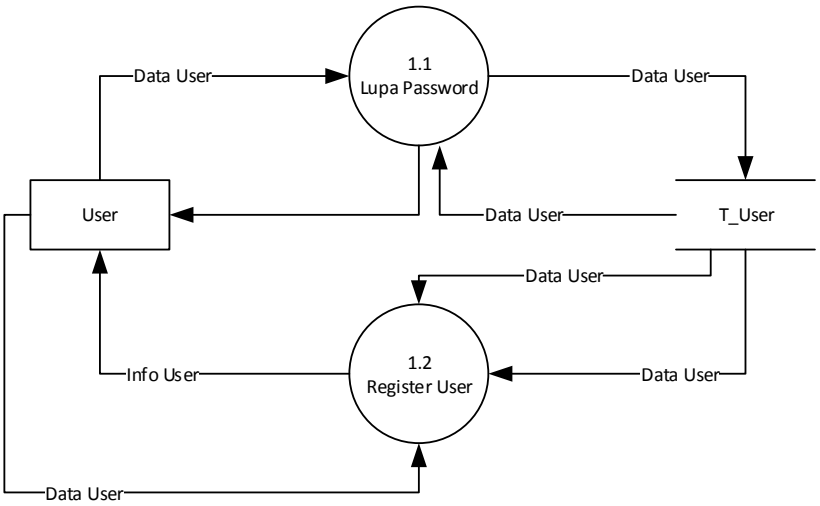
1. *Data Flow Diagram* (DFD) adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, tersruktur dan jelas. Di mana dalam DFD ini terdapat tiga jenis level yaitu diagram level 0 yang biasa dikenal dengan diagram konteks, diagram level 1 yang merupakan pemecahan dari diagram konteks dan memuat penyimpanan data, serta diagram rinci yang menguraikan proses yang ada di dalam diagram level 1. Kedalaman diagram rinci ini sendiri bersifat dinamis atau sesuai kompleksitas suatu sistem. Berikut adalah contoh dari ketiga jenis level DFD yag terimplementasi di aplikasi *play my batik.*

   Contoh diagram konteks dapat dilihat pada Gambar II.8 berikut:



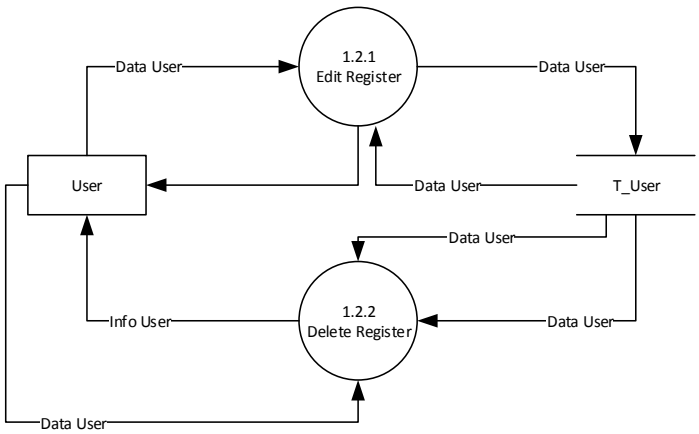
Gambar II.8 Contoh Diagram Konteks

Contoh DFD Level 1 dapat dilihat pada Gambar II.9 berikut:



Gambar II.9 Contoh DFD Level 1

Contoh DFD Level 2 merupakan rincian dari proses schedule dapat dilihat pada Gambar II.10.



Gambar II.10 Contoh DFD Level II

2. Spesifikasi proses merupakan tabel yang menggambarkan deskripsi dan spesifikasi dari setiap proses pada pemodelan DFD sesuai kebutuhan sistem. Di mana penjelasannya itu sendiri dapat berupa naratif ataupun pseudo code [16]. Berikut adalah tabel spesifikasi proses yang terdapat pada tabel Tabel II.3

Tabel II.3 Spesifikasi Proses [16]

No	Proses	Keterangan
1.	No Proses	Menyatakan nomor proses
2	Nama Proses	Menyatakan nama proses
3	*Source* (Sumber)	(Menyatakan sumber data input menuju proses)
4	*Input*	(menyatakan isi data yang masuk ke proses)
5	*Output*	(menyatakan informasi yang keluar dari proses)
6	*Destination*	(menatakan tujuan informasi *output* dari proses)
7	Logika Proses	(menyatakan algoritma dari proses)

3. Kamus data atau data *dictionary* adalah kumpulan fakta tentang data dan berbagai kebutuhan informasi dari suatu sistem informasi. Dengan kamus data analis sistem dapat mendefinisikan data yang mengalir di dalam sistem dengan lengkap. Pada tahap analisis sistem, kamus data digunakan sebagai alat komunikasi antara analis sistem dengan pemakai sistem tentang data yang mengalir ke sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem. Pada tahap perancangan sistem, kamus data digunakan untuk merancang input, merancang laporan-laporan dan database. Kamus data dibuat berdasarkan arus data yang ada di *Data Flow Diagram* (DFD). Arus data di DFD itu sendiri sifatnya global yang berarti hanya ditunjukkan nama arus datanya saja, sedangkan di dalam kamus data itu sendiri dapat dilihat keterangan yang lebih terperinci tentang struktur dari suatu arus data di DFD. Berikut adalah contoh penerapan kamus data pada Tabel II.4.

Tabel II.4 Kamus Data

Nama aliran data	Nama data dari data yang digunakan
Where used/How used	Daftar pada poin proses apa data digunakan
Keterangan	Uraian singkat dari data yang digunakan
Struktur Data	Daftar komponen data yang ada pada data yang digunakan
Deskripsi	Jenis data dalam representasi komputer untuk masing-masing data.

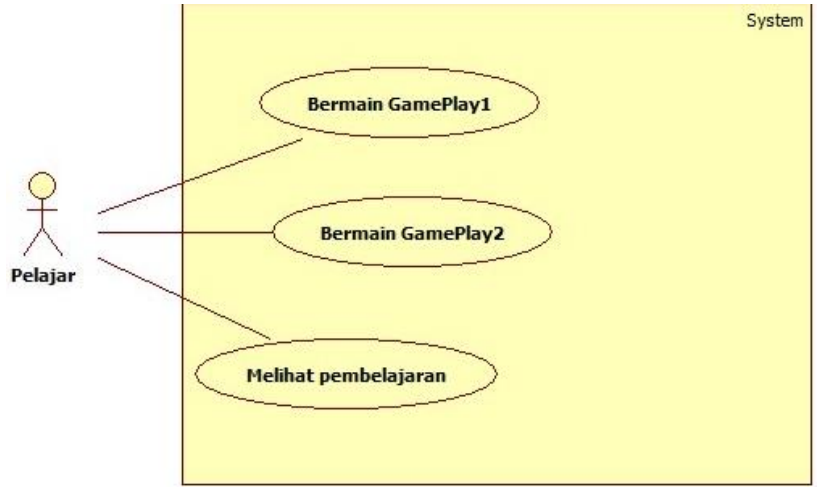
II.11 Object Orientied Analysis and Design (OOAD)

Analisis Design Berorientasi Objek adalah suatu pendekatan rekayasa perangkat lunak dari sebuah sistem yang terdiri dari sekelompok objek yang saling berinteraksi, dan setiap objek mewakili beberapa entitas yang ditandai dengan adanya sebuah kelas. Terdapat dua bagian pembentukan pada sistem jika mengguankan pendekatan OOAD yaitu *Object Oriented Analysis* mendefiniskan teknik pemodelan objek dengan mengacu pada persyaratan fungsional suatu sistem sedangkan *Object Oriented Design* memaparkan sebuah bentuk sistem berdasarkan hasil analysis OOA. Konsep dasar yang terdapat dalam sistem OOAD adalah *class, object, polymorphsm, inheritance dan encapsulation.*

UML (*Unifield Modeling Language*) adalah sebuah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan dan membangun sistem perangkat lunak berbasis Objek (*Object Oritend*). UML mulsi diperkenalkan oleh Object Management Group, sebuah organisasi yang telah mengembangkan model teknologi dan standar OOP sejak tahun 1980an. Berikut empat macam diagram UML yang sering digunakan untuk pembangunan perangakat lunak berorientasi objek

1. *Use case diagram*

   Use case diagram merupakan gambaran spesifik dari bagian peersayaratan fungsionalitas yang harus dilakukan oleh sistem, di dalam penggunaanya use case berperan sebagai petunjuk untuk berbagai elemen dan desain dari sistem yang dibangun [17]. *Use case diagram* terdiri atas diagram untuk *use case* dan a*ctor*. *Acto*r merepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi. *Use case* merepresentasikan operasi-operasi yang dilakukan oleh *actor*. *Use case* digambarkan berbentuk elips dengan nama operasi dituliskan didalamnnya. *Actor* yang melakukan operasi dihubungkan dengan garis lurus ke *use case*. Berikut adalah contoh use case yang terdapat pada Gambar II.11.

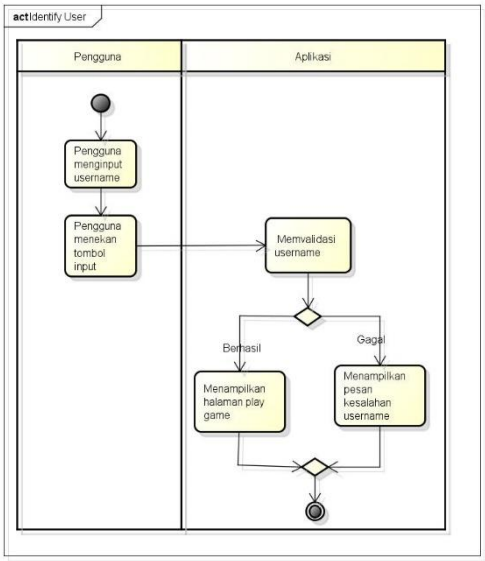


Gambar II.11 Contoh Use Case Diagram

2. *Activity Diagram*

Use case menunjukan apa yang seharusnya sistem lakukan sedangkan activity diagram memberikan penjelasan bagaimana sistem tersebut akan menyelesaikan tujuannya tersebut. *Activity diagram* menunjukan tindakan tingkat tinggi yang merantai bersama-sama untuk mewakili sebuah proses yang terjadi di dalam sistem [17].

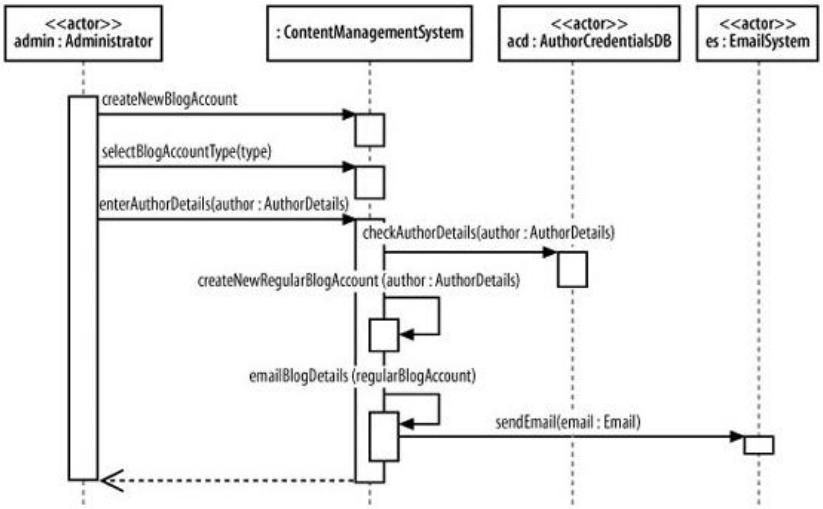
Activity Diagram ini sendiri memiliki struktur diagram yang mirip *flowchart* atau data *flow diagram* pada perancangan terstruktur. *Activity Diagram* dibuat berdasarkan sebuah atau beberapa *use case* pada *use case diagram.* Berikut adalah contoh activity diagram yang terdapat pada Gambar II.12.



Gambar II.12 Contoh Activity Diagram

3. *Sequence Diagram*

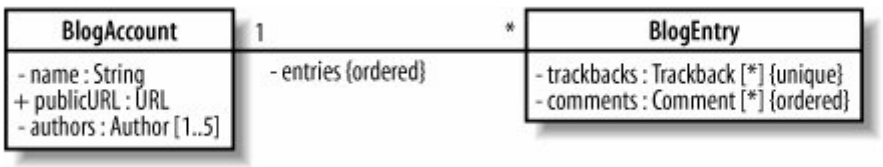
Sequence diagram menjelaskan secara detail urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari *use case*: interaksi yang terjadi antar *class*, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi [17]. Komponen utamanya adalah objek yang digambarkan dengan kotak segi empat atau bulat, *message* yang digambarkan dengan garis penuh, dan waktu yang ditunjukkan dengan *progress vertical*. Berikut adalah contoh *sequence diagram* membuat sebauh blog yang terdapat pada Gambar II.13.



Gambar II.13 Contoh Sequence Diagram

4. *Class Diagram*

Class *diagram* merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. *Class* diagram menunjukkan hubungan antar *class* dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan [17]. Berikut adalah contoh class diagram yang terdapat pada Gambar II.14.



Gambar II.14 Contoh Class Diagram

II.12 Object Relational Mapping

Pemetaan Objek Relasional (*Object Relational Mapping*) adalah teknologi yang digunakan untuk menjembatani impedansi ketidakcocokan antara program berorientasi objek dengan database relasional [18]. Hal tersebut merupakan usaha untuk mencoba menghilangkan duplikasi data sehingga meminimalisir biaya pemeliharaan dan untuk mengatasi kerentanan error terhadap penyatuan program orientasi objek dengan database relasional.

Beberapa ketidakcocokan dapat diklasifikasikan sebagai ketidaksamarataan antara fungsi dan *native-operation* bahasa pemrograman objek dengan relasi database itu sendiri. Sebagai contohnya, model objek tidak mungkin dapat menyimpan langsung kedalam database tanpa manipulasi. Hal tersebut dikarenakan database tidak memiliki kemampuan untuk menangani *inheritance* ataupun *pholymorphism*, dua hal penerapan dasar dalam pembangunan berorientasi objek. Salah satu solusi untuk memecahkan hal tersebut adalah menggunakan *tools object relational mapping.*

II.13 Web Service

Web *service* adalah aplikasi sekumpulan data (*database*), perangkat lunak (*software*) atau bagian dari perangkat lunak yang dapat diakses secara *remote* oleh berbagai perangkat dengan sebuah perantara tertentu. Secara umum, *web service* dapat dikenali dengan menggunakan URL seperti hanya web pada umumnya. Namun yang membedakan *web service* dengan web pada umumnya adalah interaksi yang diberikan oleh *web service*. Berbeda dengan URL *web* pada umumnya, URL *web service* hanya mengandung kumpulan informasi, perintah, konfigurasi atau sintaks yang berguna membangun sebuah fungsi-fungsi tertentu dari aplikasi.

Web *service* dapat diartikan juga sebuah metode pertukaran data, tanpa memperhatikan dimana sebuah *database* ditanamkan, dibuat dalam bahasa apa sebuah aplikasi yang mengkonsumsi data, dan di *platform* apa sebuah data itu dikonsumsi. Web *service* mampu menunjang interoperabilitas. Sehingga *web service* mampu menjadi sebuah jembatan penghubung antara berbagai sistem yang ada [19].

Menurut W3C *Web service architecture Working Group*, pengertian Web *service* adalah sebuah sistem *software* yang di desain untuk mendukung interoperabilitas interaksi mesin ke mesin melalui sebuah jaringan. Interface *web service* dideskripsikan dengan menggunakan format yang mampu diproses oleh mesin (khususnya WSDL). Sistem lain yang akan berinteraksi dengan *web service* hanya memerlukan SOAP, yang biasanya disampaikan dengan HTTP dan XML sehingga mempunyai korelasi dengan standar Web (Web Services Architecture Working Group, 2004) [19].

II.14 JSON

JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat *(generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemprograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999 [20]. JSON merupakan format teks yang tidak bergantung pada bahasa pemprograman apapun sehingga menjadikan JSON ideal sebagai bahasa pertukaran-data. JSON terbuat dari dua struktur sebagai berikut:

1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel *hash* (*hash table*), daftar berkunci (*keyed list*), atau *associative array*.
2. Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*sequence*).

II.15 API

API (*Application Programming Interface*) adalah kumpulan fungsi-fungsi atau perintah-perintah untuk menggantikan bahasa yang digunakan dalam *system call* dengan bahasa yang lebih terstruktur dan mudah dimengerti.

Keuntungan memprogram dengan menggunakan API adalah:

1. Portabilitas. Programmer yang menggunakan API dapat menjalankan programnya dalam sistem operasi mana saja asalkan sudah terinstall API

tersebut. Sedangkan *system call* berbeda antar sistem operasi, dengan catatan dalam implementasinya mungkin saja berbeda.

2. Lebih Mudah Dimengerti. API menggunakan bahasa yang lebih terstruktur dan mudah dimengerti daripada bahasa *system call*. Hal ini sangat penting dalam hal editing dan pengembangan.

System call interface ini berfungsi sebagai penghubung antara API dan *system call* yang dimengerti oleh sistem operasi. System call interface ini akan menerjemahkan perintah dalam API dan kemudian akan memanggil system calls yang diperlukan.

Untuk membuka suatu file tersebut *user* menggunakan program yang telah dibuat dengan menggunakan bantuan API, maka perintah dari user tersebut diterjemahkan dulu oleh program menjadi perintah *open().* Perintah *open()* ini merupakan perintah dari API dan bukan perintah yang langsung dimengerti oleh kernel sistem operasi. Oleh karena itu, agar keinginan *user* dapat dimengerti oleh sistem operasi, maka perintah *open()* tadi diterjemahkan ke dalam bentuk *system call* oleh *system call interface.* Implementasi perintah *open()* tadi bisa bermacam-macam tergantung dari sistem operasi yang kita gunakan.

II.16 Microsoft Visual Studio

Microsoft Visual Studio merupakan salah satu perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis ataupun personal, dalam bentuk aplikasi *console, windows* ataupun aplikasi web. Visual studio adalah salah satu aplikasi pemrograman yang terintegrasi dengan bahasa pemrograman lainnya atau yang *integrated developtment environment* (IDE) selain itu pada microsoft visual studio disediakan SDK, Library MSDN dan paket kompiler antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe.

Microsoft Visual Studio juga dapat digunakan untuk mengembangkan aplikasi dalam native code (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun managed code (dalam bentuk Microsoft Intermediate Language

di atas .NET Framework). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi Silverlight, aplikasi Windows Mobile (yang berjalan di atas .NET Compact Framework).