

---

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Rancang Bangun**

Rancang merupakan serangkaian prosedur untuk menerjemahkan hasil analisa dari sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem diimplementasikan (Pressman RS 2012). Rancangan sistem adalah penentuan proses dan data yang diperlukan oleh sistem baru (Leod m 2002). Perancangan adalah kegiatan yang memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternatif sistem yang terbaik (Ladjamudin 2005). Sedangkan pengertian bangun atau pembangunan sistem adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun sebagian (Pressman RS 2012).

#### **2.2 Game**

*Game* bermakna “Permainan”.Teori permainan adalah suatu cara belajar yang digunakan dalam menganalisa sejumlah pemain maupun perorangan yang menunjukkan strategi-strategi rasional. Teori permainan pertama kali ditemukan oleh sekelompok ahli matematika pada tahun 1944. Teori itu yang berisi: “Permainan terdiri atas sekumpulan peraturan yang membangun situasi bersaing”. Menurutnya permainan terdiri atas sekumpulan peraturan yang membangun situasi bersaing dari 2 atau beberapa orang kelompok dengan memilih strategi yang dibangun untuk memaksimalkan kemenangan sendiri ataupun untuk meminimalkan kemenangan lawan.

Dalam *game* ini juga perlu adanya sekenario agar alur permainan pun jelas dan terarah. Sekenario di sini bisa meliputi *setting map*, level, alur cerita, bahkan efek yang ada dalam *game* tersebut. Suatu cara belajar yang digunakan dalam menganalisa

interaksi antara sejumlah pemain maupun perorangan yang menunjukkan strategi-strategi yang rasional.

Menurut (Muhammad As'adi,2009) mengatakan bahwa definisi permainan adalah: usaha olah diri (olah pikiran dan olah fisik) yang sangat bermanfaat bagi peningkatan dan pengembangan motivasi, kinerja dan prestasi dalam melaksanakan tugas dan kepentingan organisasi dengan lebih baik.

Menurut (Andang Ismail,2009) mendefinisikan permainan sebagai suatu aktifitas yang membantu anak mencapai perkembangan yang utuh, baik fisik, intelektual, sosial, moral, dan emosional.

### **2.2.1 *Game* Edukasi**

Menurut (Handriyantini,2009) *Game* Edukasi adalah salah satu jenis media yang digunakan dalam memberikan pengajaran yang berupa permainan dengan tujuan untuk merangsang daya pikir dan meningkatkan konsentrasi melalui media yang unik dan menarik. Pengertian ini tentu saja mengidentifikasikan bahwa *Game* edukasi bertujuan untuk menunjang proses belajar mengajar dengan kegiatan yang menyenangkan dan lebih kreatif.

Menurut (Marc Prensky,2012) *Game* edukasi adalah suatu bentuk permainan (game) yang didesain atau dibuat untuk tujuan belajar, akan tetapi dalam *game* edukasi biasanya menawarkan bermain untuk bersenang-senang.

Menurut (Novia Desta,2016) Arti *Game* edukasi adalah permainan yang dibuat dengan tujuan pembelajaran yang bukan hanya bermaksud menghibur sehingga diharapkan bisa menambah wawasan pengetahuan.

### 2.3 *Autisme*

*Autisme* berasal dari kata *autos* yang berarti aku. Pada pengertian non ilmiah kata tersebut dapat ditafsirkan bahwa semua anak yang mengarah pada dirinya sendiri disebut dengan autisme (Yuwono,2009). Autisme bukan suatu gejala penyakit tetapi berupa sindroma (kumpulan gejala) dimana terjadi penyimpangan perkembangan sosial, kemampuan berbahasa dan kepedulian terhadap sekitar. Autisme adalah suatu keadaan dimana seseorang anak berbuat semaunya sendiri baik secara berpikir maupun berperilaku. Keadaan ini terjadi sejak usia 2-3 tahun tanpa memandang sosial ekonomi mapan maupun kurang, atau atau dewasa dan semua etnis (Yatim 2002).

*Autisme* sebenarnya bukan hal baru dan sudah ada sejak lama, namun belum terdiagnosis sebagai autis. Menurut cerita terdahulu sering kali ada anak yang dianggap aneh, anak tersebut sering kali menunjukkan gejala yang tidak biasa. Mereka menolak bila digendong, menangis kalau malam dan tidur bila siang hari. Mereka sering kali bicara sendiri dengan bahasa yang sering tidak dimengerti oleh orang tuanya. Apabila dalam kondisi marah mereka bisa menggigit, mencakar, menjambak atau menyerang.

Para peneliti kemudian menyatakan bahwa autisme bukan hanya gangguan fungsional, tetapi didasari adanya gangguan organik dalam perkembangan otak (Nugraheni, 2008).

Menurut (McCandless,2003) *Autistic Spectrum Disorder* (ASD) adalah suatu grup gangguan perkembangan anak yang berkisar dari autisme klasik seperti *Attention Deficit Disorder* (ADD), *Attantion Deficit Hyperactivity Disorder* (ADHD) dan *Perpasive Developmental Disorder* (PDD). PDD adalah diagnosis yang diberikan kepada anak-anak apabila mereka tidak mencapai tonggak-tonggak seperti seharusnya dan menunjukkan gejala-gejala autisme. Anak tersebut memiliki sedikit kemampuan untuk berbicara dan berkomunikasi. Seorang anak yang

didiagnosa dengan ADD memiliki kesulitan dalam mempertahankan kemampuan memusatkan perhatiannya. Seorang anak hiperaktif dengan ADD diberi label ADHD. Ditingkat lebih atas dari spectrum autis adalah *Asperger's Syndrome*.

Istilah ini digunakan untuk mendeskripsikan seorang anak autistik yang dapat berfungsi pada tingkat yang lebih tinggi. Anak-anak Asperger merupakan anak-anak yang luar biasa cerdas. Mereka menggunakan dan mengerti perbendaharaan kata secara luas, tetapi mereka memiliki minat yang sangat sempit dan menunjukkan banyak kekurangan dari segi sosial. Seorang anak dengan *Asperger's Syndrome* bisa sangat ahli mengenai mesin cuci, tapi mesin cuci adalah satu-satunya hal yang mau ia bicarakan.

Menurut (Mujiyanti,2011) ada banyak tingkah laku yang tercakup dalam anak autis dan ada 4 gejala yang selalu muncul yaitu :

1. Isolasi sosial

Banyak anak autis yang menarik diri dari kontak sosial kedalam suatu keadaan yang disebut *extreme autistic alones*. Hal ini akan semakin terlihat pada anak yang lebih besar, dan ia akan bertingkah laku seakan-akan orang lain tidak ada.

2. Kelemahan kognitif

Anak autis sebagian besar ( $\pm 70\%$ ) mengalami retardasi mental ( $IQ < 70$ ) disebut dengan autis dengan tuna grahita tetapi anak autis infertil sedikit lebih baik, contohnya dalam hal yang berkaitan dengan hal sensor motorik. Anak autis dapat meningkatkan hubungan sosial dengan temannya, tetapi hal itu tidak berpengaruh terhadap retardasi mental yang dialami.

3. Kekurangan dalam bahasa

Lebih dari setengah autis tidak dapat berbicara, yang lainnya hanya mengoceh, merengek, atau menunjukkan *ecocalia*, yaitu menirukan apa yang dikatakan orang lain. Beberapa anak autis mengulang potongan lagu, iklan TV atau potongan kata yang terdengar tanpa tujuan.

#### 4. Tingkah laku stereotif

Anak autis sering melakukan gerakan yang berulang-ulang secara terus menerus tanpa tujuan yang jelas. Seperti berputar-putar, berjingkat-jingkat dan lain sebagainya. Gerakan ini dilakukan berulang-ulang disebabkan karena kerusakan fisik, misalnya ada gangguan neurologis. Anak autis juga mempunyai kebiasaan menarik-narik rambut dan menggigit jari. Walaupun sering kesakitan akibat perbuatannya sendiri, dorongan untuk melakukan tingkah laku yang aneh ini sangat kuat dalam diri mereka. Anak autis juga hanya tertarik pada bagian-bagian tertentu dari sebuah objek misalnya pada roda mobil-mobilan. Anak autis juga menyukai keadaan lingkungan dan kebiasaan yang monoton.

### 2.3.1 Jenis-jenis Autis

Menurut (Yatim,2002) autisme terdiri dari 3 jenis yaitu persepsi, reaksi dan yang timbul kemudian.

#### 1. Autis persepsi

Autis persepsi merupakan autisme yang timbul sebelum lahir dengan gejala adanya rangsangan dari luar baik kecil maupun besar yang dapat menimbulkan kecemasan. Misalnya pada ibu hamil yang mempunyai genetik autisme dia mempunyai kecemasan akan menurun terhadap janin yang dikandungnya.

#### 2. Autis reaktif

Autisme reaktif ditunjukkan dengan gejala berupa penderita membuat gerakan-gerakan tertentu yang berulang-ulang dan kadang disertai kejang dan dapat diamati pada anak usia 6-7 tahun. Anak memiliki sifat rapuh dan mudah terpengaruh pada dunia luar.

#### 3. Autis yang timbul kemudian

Jenis autisme ini diketahui setelah anak agak besar dan akan kesulitan dalam mengubah perilakunya karena sudah melekat atau ditambah adanya pengalaman yang baru atau gejala autis terlihat saat anak mulai dewasa.

Menurut (McCandless,2003) autis dibagi menjadi dua, yaitu :

#### 1. Autisme klasik

Autis sebelum lahir merupakan bawaan yang diturunkan dari orang tua ke anak yang dilahirkan atau sering disebut autis yang disebabkan oleh genetika (keturunan). Kerusakan saraf sudah terdapat sejak lahir, karena saat hamil ibu terinfeksi virus seperti *rubella*, atau terpapar logam berat berbahaya seperti merkuri dan timbal yang berdampak mengacaukan proses pembentukan sel-sel otak janin.

#### 2. Autisme regresif

Muncul saat anak berusia 12 sampai 24 bulan. Sebelumnya perkembangan anak relatif normal, namun sejak usia anak 2 tahun perkembangannya merosot. Anak yang tadinya sudah biasa membuat kalimat beberapa kata berubah menjadi diam dan tidak lagi berbicara. Anak menjadi acuh dan tidak ada lagi kontak mata. Kalangan ahli menganggap autisme regresif karena anak terkontaminasi langsung faktor pemicu. Paparan logam berat terutama merkuri dan timbal dari lingkungan merupakan faktor yang paling disorot.

### 2.3.2 Klasifikasi Autis

Autisme dapat diklasifikasikan menjadi beberapa bagian berdasarkan gejalanya. Sering kali pengklasifikasian disimpulkan setelah anak didiagnosa autis. Klasifikasi ini dapat diberikan melalui *Childhood Autism Rating Scale*(CARS). Pengklasifikasiannya adalah sebagai berikut :

#### 1. Autis Ringan

Pada kondisi ini anak autis masih menunjukkan adanya kontak mata walaupun tidak berlangsung lama. Anak autis ini dapat memberikan sedikit respon ketika dipanggil namanya, menunjukkan ekspresi-ekspresi muka, dan dalam berkomunikasi dua arah meskipun terjadinya hanya sesekali.

## 2. Autis Sedang

Pada kondisi ini anak autis masih menunjukkan sedikit kontak mata namun tidak memberikan respon ketika namanya dipanggil. Tindakan agresif atau hiperaktif, menyakiti diri sendiri, acuh, dan gangguan motorik yang stereotipik cenderung agak sulit untuk dikendalikan tetapi masih bisa dikendalikan.

## 3. Autis Berat

Anak autis yang berada pada kategori ini menunjukkan tindakan-tindakan yang sangat tidak terkendali. Biasanya anak autis memukul-mukulkan kepalanya ke tembok secara berulang-ulang dan terus menerus tanpa henti. Ketika orang tua berusaha mencegah, namun anak tidak memberikan respon dan tetap melakukannya, bahkan dalam kondisi berada di pelukan orang tuanya, anak autis tetap memukul-mukulkan kepalanya. Anak baru).

## 2.4 *Android*

(Safaat,2012) Mendefinisikan Android adalah sistem operasi yang berbasis Linux untuk telepon seluler seperti telepon pintar (*smartphone*) dan komputer tablet. Sistem operasi ini identik dengan lambang robot hijau. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Karena merupakan *software* berbasis *open source* maka bagi seorang pengembang (*developer*) dapat memanfaatkannya secara gratis dengan melakukan perubahan dan pengembangan terhadap aplikasi yang akan digunakan. Sedangkan bagi produsen ponsel, keuntungan yang di peroleh yaitu berupa *efisiensi* karena produsen dapat memangkas biaya produksi.

### 2.4.1 Versi dan Jenis-Jenis *Android*

Pengembangan Android dimulai dengan berdirinya Android, Inc. Pada oktober 2003 dengan tujuan membuat *mobile device* yang lebih *smart* untuk menyaingi *Symbian* dan *Windows Mobile* yang populer saat itu (*Iphone* dan *Blackberry* belum diliris). Pada tahun 2005, *Android Inc.* Diakuisisi oleh *google*.

Pegembangan terus dilanjutkan sampai android versi beta diluncurkan pada tanggal 5 November 2007 bersamaan dengan berdirinya OHA (Open Handset Alliance). Versi dan jenis-jenis dijelaskan pada tabel 2.1.

VERSI	NAMA	RILIS	CATATAN
4.4	Kit Kat	Oktober 2013	Diumumkan September 2013
5.0	Lollipop	2014	Peningkatan system keamanan dan tampilan
6.0	Marshmallow	28 Mei 2015	

#### 2.4.2 Kelebihan *Android*

Android memiliki kelebihan pada beberapa fitur diantaranya adalah :

1. *User Friendly*, kalimat ini sangat melekat pada sistem operasi Windows milik microsoft, ibaratnya kita dengan sangat mudah mengoprasikan komputer hanya dengan belajar beberapa hari bahkan beberapa jam saja, dan ini juga melekat pada Android yang berjalan pada *Smartphone*.
2. *Notifications*, kita dengan sangat mudah mendapatkan notifikasi dari *smartphone android* dengan mengatur beberapa akun *Email*, *SMS*, *Voice Dial*, *Update* dan lain sebagainya.
3. Tampilan, dari segi tampilan *Android* tidak kalah bagusya dari *iOs* milik *Apple*, karena memang dari awal *android* hampir mengusung teknologi *iOs* hanya saja ini versi murahya.
4. *Open Source*, *Operating system* ini memang dibuat *open source* oleh penciptanya, karena memang berbasis *kernel Linux*. Jadi jangan kaget jika diluar sana ada banyak sekali *Custom Rom* untuk masing-masing perangkat *android*.
5. Aplikasi, akan disajikan jutaan pilihan aplikasi yang menarik dari yang gratis hingga berbayar, dan kita bisa mendownloadnya di Google Play.



### 2.4.3 Kekurangan *Android*

Kekurangan *Android* adalah :

1. *Update System*, untuk melakukan *update system* tidaklah mudah kita harus menunggu dari masing-masing *vendor* untuk merilis *Update Versi* yang terbaru.
2. Baterai cepat habis, ini sering terjadi jika kita menyalakan paket data dan menggunakan *widget* serta aplikasi yang berjalan secara berlebihan, untuk mengatasinya kita harus mengurangi aktivitas aplikasi pada *smartphone*.
3. Lemot atau Lag, ini ada kaitannya dengan spesifikasi dari masing-masing perangkat, namun ada kalanya *Android* tidak bersahabat dengan beberapa *aplikasi* dikarenakan Ram ataupun prosesor yang kurang memadai, jadi sesuaikan aplikasi dengan perangkatnya.

## 2.5 Perangkat Lunak Yang Digunakan

### 2.5.1 *Adobe Flash Professional CS6*

(Andi,2013) mendefinisikan bahwa *Adobe Flash Professional CS6* merupakan perangkat lunak multiguna yang dapat dimanfaatkan untuk berbagai macam animasi dengan berbagai fitur canggih yang ada didalamnya dan juga dapat menggambar, membuat animasi, hingga digunakan untuk membuat *game* yang menarik dan berkualitas. Area kerja *Adobe Flash Professional CS6* dirancang secara khusus agar ruang kerja yang digunakan dapat diatur dan lebih mudah dipahami oleh pemakai pemula maupun para desainer *flash* yang telah berpengalaman.

*Adobe Flash Professional CS6* adalah program yang cukup kompleks sehingga tidak mungkin untuk menjabarkannya secara lengkap. Disini hanya akan diberikan sedikit pengertian dan fungsi dasar dari *Adobe Flash Professional CS6*.

### 2.5.2 *Adobe Integrated Runtime*

*Adobe Integrated Runtime* juga dikenal sebagai *Adobe AIR* adalah sistem *runtime cross-platform* yang dikembangkan oleh *Adobe Systems* untuk membangun aplikasi

*desktop* dan aplikasi *mobile*, diprogram menggunakan *Adobe Flash*, *Action Script* dan opsional *Adobe Flex*. *Runtime* yang mendukung aplikasi diinstal pada *Windows*, *OS X* dan sistem operasi *mobile* seperti *Android*, *iOS* dan *BlackBerry Tablet OS*. Hal ini juga awalnya berjalan pada *Linux*, namun dukungan dihentikan pada versi 2.6 pada tahun 2011.

### **2.5.3 Adobe Photoshop**

(Henky Prihatna, 2005) Mendefinisikan *Photoshop* adalah salah satu *software* pengolah *grafik* yang banyak digunakan oleh para *desainer grafis* dan *web* di seluruh dunia. Tampilannya yang mudah dipahami, kelengkapan fasilitas yang ditawarkan, serta kemudahan memperoleh fasilitas pendukung dari berbagai sumber menjadikan *Photoshop* menjadi pilihan paling handal bagi para *desainer*.

## **2.6 UML (*Unified Modeling Language*)**

(Sulistyorini. P, 2009) Mendefinisikan *Unified Modelling Language* (UML) adalah sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML dapat dibuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek seperti C++, *Java*, atau VB. NET.

### **2.6.1 Diagram UML**

Model-model diagram dapat dikelompokkan berdasarkan sifatnya yaitu statis dan dinamis. Jenis diagram tersebut antara lain:

1. Diagram Kelas

Bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, dan relasi-relasi.

2. Diagram Paket

Bersifat statis. Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian dari diagram komponen.

3. Diagram *Use case*

Bersifat statis. Diagram ini memperlihatkan himpunan *use-case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna. *Elemen use case* dapat dilihat pada table 2.2.

4. Diagram Interaksi atau *Sequence*

Bersifat dinamis. Diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.

5. Diagram Komunikasi

Bersifat dinamis. Diagram sebagai pengganti diagram kolaborasi UML yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.


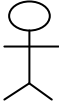

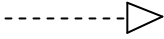
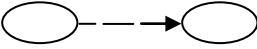
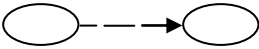
6. Diagram *Statechart*

Bersifat dinamis. Diagram ini memperlihatkan keadaan-keadaan setiap sistem, memuat status, transisi, kejadian serta aktivitas.

7. Diagram Aktivitas

Bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek. Pada tabel 2.2 merupakan simbol-simbol yang ada pada diagram aktivitas.

**Tabel 2.2** *Elemen Use Case*

<b>SIMBOL</b>	<b>KETERANGAN</b>
<i>Use Case</i> 	Menggambarkan bagaimana seseorang akan menggunakan atau memanfaatkan sistem.
Aktor 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri.
Asosiasi 	Komunikasi antara <i>use case</i> dan aktor yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki
Generalisasi 	Sebagai penghubung antara aktor- <i>use case</i> atau <i>use case-use case</i> .
<<Include>> 	<i>Include Relationship</i> (relasi cakupan) : Memungkinkan suatu <i>use case</i> untuk menggunakan fungsionalitas yang disediakan oleh <i>usecase</i> yang
<<Extend>> 	<i>Extend Relationship</i> : Memungkinkan relasi <i>use case</i> memiliki kemungkinan untuk memperluas fungsionalitas yang disediakan oleh

## 8. Diagram Komponen

Bersifat statis. Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem pada komponen-komponen yang telah ada sebelumnya.

## 9. Diagram *Deployment*

Bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan. Memuat simpul-simpul beserta komponen-komponen yang ada di dalamnya.

Keunggulan menggunakan UML dibandingkan menggunakan metodologi terstruktur:

1. *Uniformity*

Pengembangan cukup menggunakan satu metodologi dari tahap analisis hingga perancangan. Memungkinkan merancang komponen antarmuka secara terintegrasi bersama perancangan perangkat lunak dan perancangan struktur data.

2. *Understandability*

Kode yang dihasilkan dapat diorganisasi kedalam kelas-kelas yang berhubungan dengan masalah yang sesungguhnya sehingga lebih mudah untuk dipahami.

3. *Stability*

Kode program yang dihasilkan relatif stabil sepanjang waktu, karena mendekati permasalahan yang sesungguhnya.

4. *Reusability*

Dengan metodologi berorientasi objek, dimungkinkan penggunaan ulang kode, sehingga pada akhirnya akan sangat mempercepat waktu pengembangan perangkat lunak (atau sistem informasi).

## 2.7 Metode *Prototype*

### 2.7.1 Pengertian Metode *Prototype*

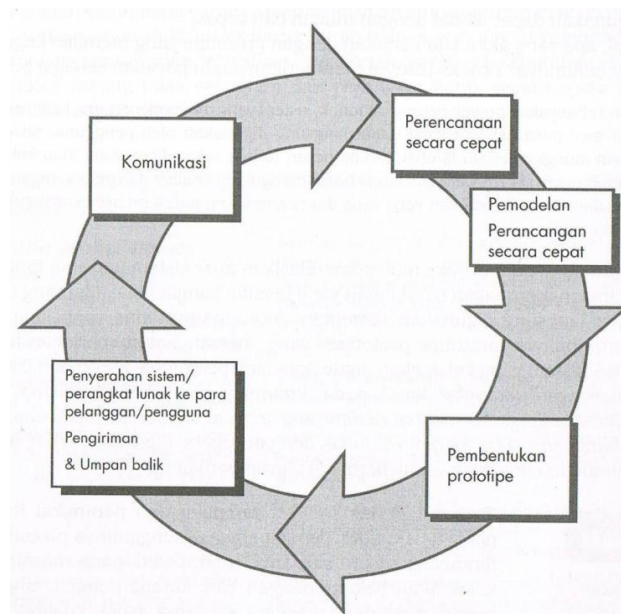
(Pressman,2002) Mendefinisikan *prototype* paradigma merupakan pelanggan yang efektif dalam merancang perangkat lunak *Prototype* model dimulai dengan mengumpulkan kebutuhan. Pengembang dan pelanggan bertemu dan mendefinisikan objek keseluruhan dari perangkat lunak, mengidentifikasi segala kebutuhan yang diketahui dan kemudian melakukan “perancangan kilat”. Perancangan kilat berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan nampak bagi pelanggan atau pemakai (contohnya pendekatan input dan format output).

Perancangan kilat membawa kepada konstruksi sebuah *prototype*. *Prototype* tersebut dievaluasi oleh pelanggan dan dipakai untuk menyaring kebutuhan pengembangan perangkat lunak. *Prototype* paradigma dapat dilihat pada gambar 2.4 berikut :

## 2.7 Metode Pengembangan *Prototype*

(Pressman 2012), menyatakan bahwa *prototyping* merupakan metode yang efektif dalam merancang perangkat lunak. *Prototyping* dimulai dengan mengumpulkan kebutuhan yang akan dirancang. Pengembang mendefinisikan *object* keseluruhan dari perangkat lunak, mengidentifikasi segala aktifitas yang diketahui dan kemudian melakukan “perancangan kilat”. Perancangan kilat berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan nampak bagi pelanggan atau pemakai (contohnya pendekatan *input* dan format *output*).

Perancangan kilat membawa kepada konstruksi sebuah *prototype*. *Prototype* tersebut dievaluasi oleh pelanggan dan dipakai untuk menyaring kebutuhan pengembangan perangkat lunak. Adapun tahapan metode pengembangan *prototype* adalah sebagai berikut :



**Gambar 2.3** *Prototype*

*Prototype* juga dapat didefinisikan sebagai proses pengembangan suatu *prototype* secara cepat untuk digunakan terlebih dahulu dan ditingkatkan terus menerus sampai

didapatkan sistem yang utuh. *Prototyping* merupakan proses yang digunakan untuk membantu pengembang perangkat lunak dalam membentuk *prototype* dari perangkat lunak yang harus dibuat.

### **2.7.2 Kelebihan dan Kekurangan Metode *Prototype***

Kelebihan *prototype* adalah :

1. Adanya komunikasi yang baik antara pengembang dan pelanggan.
2. Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan.
3. Pelanggan berperan aktif dalam pengembangan sistem.
4. Lebih menghemat waktu dalam pengembangan sistem.
5. Penerapan menjadi lebih mudah karena pemakai mengetahui apa yang diharapkannya.

Sedangkan kekurangan *prototyping* adalah :

1. Pelanggan tidak melihat bahwa perangkat lunak belum mencerminkan kualitas perangkat lunak secara keseluruhan dan belum memikirkan pemeliharaan dalam jangka waktu yang lama.
2. Pengembang biasanya ingin cepat menyelesaikan proyek sehingga menggunakan algoritma dan bahasa pemrograman sederhana.
3. Hubungan pelanggan dengan komputer mungkin tidak menggambarkan teknik perancangan yang baik.

## **2.8 StoryBoard**

(Diartono, 2008) Mendefinisikan *Storyboard* merupakan suatu pemetaan elemen-elemen multimedia dalam setiap layar program multimedia. *Storyboard* ini merupakan panduan bagi para *programmer* dan *graphics designer* dalam membangun suatu proyek multimedia, karena dalam *storyboard* ini digambarkan mengenai elemen-elemen apa saja yang digunakan dalam setiap rancangan layar yang akan dibangun.

*Storyboard* juga berfungsi sebagai media penghubung antara *programmer design* dan *client* sebelum aplikasi dibangun.

Sebuah *storyboard* harus terdiri dari:

1. Nama proyek, modul dan nomor halaman atau nomor layar.
2. Gambar *sketsa* layar atau halaman dalam layar, gambar, animasi, suara, narasi, video, warna, penempatan, serta ukuran gambar.