

## BAB II

### LANDASAN TEORI

#### 2.1 Konsep Dasar Sistem Informasi

Sistem informasi termasuk komponen utama yang harus ada dalam penelitian ini. Sebelum membangun sistem, sebaiknya kita memahami terlebih dulu konsep sistem informasi tersebut. Berikut dijelaskan konsep dasar sistem informasi:

##### 2.1.1 Pengertian Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa *subsystem* atau bagian-bagian dari sistem. Setiap sistem tidak peduli betapapun kecilnya, selalu mengandung komponen-komponen atau *subsystem-subsystem*. Setiap *sub-system* mempunyai sifat-sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan (Jogiyanto, 2005).

##### 2.1.2 Pengertian Informasi

Informasi ibarat darah yang mengalir didalam tubuh suatu organisasi, sehingga informasi ini sangat penting di dalam suatu organisasi. Informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (*event*) yang nyata (*fact*) yang digunakan untuk pengambilan keputusan (Jogiyanto, 2005).

#### 1. Siklus Informasi

Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat

**Hak Cipta Diindungi Undang-Undang**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

sejumlah data kembali. Data tersebut akan ditangkap sebagai *input*, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus. Siklus ini oleh John Burch disebut dengan siklus informasi (*Information Cycle*) atau ada yang menyebutnya dengan istilah siklus pengolahan data (*data processing cycles*) (Jogiyanto, 2005).

## 2. Kualitas Informasi

Menurut Jogiyanto (2005), kualitas dari suatu informasi tergantung dari tiga hal, yaitu:

- a. Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya.
- b. Tepat pada waktunya, berarti informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan didalam pengambilan keputusan. Bila pengambilan keputusan terlambat, maka dapat berakibat fatal untuk organisasi.
- c. Relevan, berarti informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda. Misalnya informasi mengenai sebab-musabab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan bila ditujukan kepada ahli teknik perusahaan.

## 3. Nilai Informasi

Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Kegunaan informasi adalah untuk mengurangi hal ketidakpastian didalam proses pengambilan keputusan tentang sesuatu keadaan (Jogiyanto, 2005).

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

### 2.1.3 Pengertian Sistem Informasi

Sistem informasi dapat didefinisikan sebagai suatu sistem didalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan yang cerdas (Jogiyanto, 2005).

Sistem informasi dapat terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building block*), yang terdiri dari blok masukan (*input block*), blok model (*model block*), blok keluaran (*output blok*) dan blok teknologi (*technology blok*), blok dasar data (*database block*), dan blok kendali (*control block*) (Jogiyanto, 2005).

## 2.2 Konsep Analisis, Perancangan dan Pembangunan Sistem Informasi

Terdapat beberapa alur dalam kegiatan penelitian ini, mulai dari analisa hingga pembangunan sistem informasi. Berikut dijelaskan tiap alur penelitiannya:

### 2.2.1 Analisis Sistem Informasi

Analisis permasalahan adalah hal yang pertama kita lakukan setelah kita mendapatkan spesifikasi kebutuhan pengguna (SRS). Analisis bertujuan untuk mendapatkan pemahaman secara keseluruhan tentang sistem yang akan kita kembangkan berdasarkan masukan dari calon pengguna (kadang ditambah dengan pendapat atau masukan dari beberapa pihak: pengembang, pihak manajemen, dan sebagainya) (Nugroho, 2010).

Analisis dilakukan dengan menggambarkan 3 aspek dari suatu objek: struktur statis (model objek). Struktur dinamis yang menggambarkan urutan interaksi (baik antara pengguna dengan sistem atau perangkat lunak maupun interaksi *internal* dalam sistem atau perangkat lunak itu sendiri) serta transformasi data yang coba digambarkan dengan model-model *funksional* (DAD atau Diagram Aliran Data, spesifikasi proses, dan sebagainya). (Nugroho, 2010)



#### Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Pada umumnya, analisis tidak bisa langsung jadi serta sempurna pada satu kali pemodelan. Iterasi kadang diperlukan demi pemahaman yang lebih mendalam untuk suatu permasalahan yang timbul di dunia nyata. Kadang juga terjadi perubahan pemodelan saat calon pengguna *member* tambahan masukan (Nugroho, 2010).

Definisi analisis sistem (Jogiyanto, 2005) adalah penguraian suatu sistem informasi yang utuh ke dalam bagian-bagian komponen dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya. Adapun langkah-langkah dasar yang harus dilakukan oleh analisis system adalah sebagai berikut: (Jogiyanto, 2005)

1. *Identify*, yaitu mengidentifikasi masalah.
2. *Understand*, yaitu memahami kerja dari sistem yang ada.
3. *Analyze*, yaitu menganalisis sistem.
4. *Report*, yaitu membuat laporan hasil analisis.

### 2.2.2 Perancangan Sistem Informasi

Selama analisis, perhatian kita adalah pada apa yang harus dikerjakan, terlepas dari bagaimana semuanya akan dikerjakan. Selama perancangan, keputusan dibuat tentang bagaimana pemecahan masalah akan dikerjakan. Pertama pada sistem dengan peringkat yang lebih tinggi kemudian secara bertahap ke sistem yang memiliki peringkat lebih rendah (Nugroho, 2010).

Perancangan sistem adalah tahap awal dimana pendekatan awal untuk menyelesaikan masalah yang dipilih. Selama perancangan sistem struktur keseluruhan diputuskan. Arsitektur sistem adalah suatu cara pengorganisasian sistem ke dalam apa yang dinamakan *sub* sistem-*sub* sistem (Nugroho, 2010).

#### 1. Estimasi Perangkat Keras Yang Dibutuhkan

Estimasi perangkat keras yang dibutuhkan pada dasarnya adalah mengoptimalkan kinerja dengan kendala biaya yang dimiliki. Pada dasarnya ada



Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

beberapa komponen perangkat keras yang sangat mempengaruhi kinerja suatu sistem informasi, yaitu: (Nugroho, 2010)

1) CPU (*Central Processing Unit*)

CPU adalah ‘otak’ computer: baik jika komputer tersebut berfungsi sebagai *workstation* maupun *server*. Tanpa cpu tidak mungkin kita mengharapkan suatu pekerjaan akan dapat dikerjakan oleh komputer.

2) *Memory (Random Access Memory)*

Setiap aplikasi, pada saat pemroses oleh *cpu* dilakukan, akan diletakkan dalam memori komputer yang bersifat sementara (*volatile*) sehingga idealnya kita perlu memperbesar kapasitas terpasang memori pada komputer kita secara maksimal.

3) *Layar Monitor*

Idealnya, tentu saja, kita akan memasang adapter terbaik serta *display* yang sesuai pada setiap *workstation* kita, namun sekali lagi kita terbentur masalah biaya sehingga pertimbangan kita lebih pada apa yang akan ditampilkan pada layar monitor oleh aplikasi kita.

4) *Hardisk*

Pada dasarnya setiap teknologi baru menambahkan kapasitas penyimpanan data serta meningkatkan keandalannya (dalam menangani kesalahan atau *error* yang mungkin terjadi) serta kinerjanya. Selain kapasitas yang saat ini diukur dengan satuan GB (*Giga Byte*), untuk kerja *hardisk* juga ditentukan oleh laju *transfer* datanya yang diukur dengan satuan MBPS (*Mega Byte Per Second*) yang semakin besar nilainya semakin baik serta waktu pengaksesan (*Access Time*) yang di ukur dengan MS (*Mili Second*) yang tentu saja nilainya semakin kecil semakin baik.

5) *Peripheral Tambahan (Mouse, Disk Drive, CD ROM)*

*Mouse* penting sekali saat ini karena kebanyakan aplikasi saat ini adalah aplikasi berbasis GUI (*Graphical User Interface*). Sistem operasi yang populer saat ini (*MS Windows* serta *Macintosh*) mutlak membutuhkan *mouse* sehingga *mouse* harus ada pada sistem kita.

Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

*Disk drive* penting saat kita ingin menyimpan *file-file* pada *disket*. Walaupun sistem jaringan memungkinkan kita untuk berbagi *file* antar komputer dalam jaringan, suatu saat kita mungkin ingin menyimpan file ditempat lain yang tidak terkait pada jaringan.

Saat ini kebanyakan perangkat lunak disimpan dalam CD-ROM (*Compact Disc-Read Only Memory*) sehingga sistem kita mutlak memiliki CD-ROM *Drive* (terutama untuk menginstal perangkat-perangkat lunak yang diperlukan kedalam sistem , baik sistem operasi maupun perangkat lunak aplikasi).

6) UPS (*Uninterruptible Power Supply*) dan *Stabilizer*

Naiknya-turunnya tegangan listrik (*brown-out*) sering terjadi sehingga ada bainya sistem kita memiliki *stabilizer* sebagai pelindung. UPS diperlukan jika suatu saat listrik padam dan kita sedang bekerja. UPS melindungi kehilangan data saat listrik padam dengan cara memberi waktu pada kita untuk menyimpan dan mengerjakan beberapa pekerjaan selama beberapa waktu.

7) *Modem*

Pada jaringan komputer skala luas, WAN (*Wide Area Network*) misalnya, modem mutlak diperlukan sebab tidak efisien dan tidak efektif untuk menghubungkan jarak yang sangat jauh dengan merentangkan kabel pribadi antar komputer. Untuk kasus seperti ini, lebih efektif kita menggunakan kabel telepon yang saat ini jangkauaannya sudah sangat luas.

8) *Printer*

Disuatu saat dimasa depan kelak, orang berpikir tentang lingkungan kerja yang bebas kertas karena suatu saat mungkin sumber daya alam untk membuat kertas (kayu misalnya) akan semakin sulit didapatkan. Meski begitu, saat ini laporan-laporan, faktur-faktur, dokumen-dokumen, dan sebagainya masih dicetak dalam kertas karena itulah saat ini kita masih membutuhkan *printer*.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

## 2. Pemilihan Sistem Operasi

Sistem operasi merupakan bagian yang sangat penting pada suatu sistem informasi berbasis komputer karena sistem operasi menangani kerja tiap komponen perangkat keras, misalnya: CPU, RAM, *Disk Drive*, *Hardisk*, sarana komunikasi dengan komputer lain dengan NIC atau *Network Interface Card*, dan sebagainya, sehingga bisa berfungsi seperti yang diharapkan. Sistem operasi untuk sistem informasi berbasis jaringan komputer umumnya dapat dibagi menjadi 2, yaitu: (Nugroho, 2010)

### 1) Sistem Operasi *Client*

Pertimbangan dalam memilih sistem operasi *client* adalah kemudahan dalam penggunaan (*usability*), spesifikasi perangkat keras *client*, topologi logika jaringan, serta biaya. Misalkan jika kita punya spesifikasi perangkat keras *client* yang rendah, juga biaya pembelian sistem operasi yang rendah, mungkin kita perlu mempertimbangkan penggunaan DOS sebagai sistem operasi (dengan resiko tingkat kemudahan penggunaan rendah). Pada kasus lain, kita punya spesifikasi perangkat keras yang cukup, biaya yang cukup, serta topologi jaringan *peer-to-peer*, serta menginginkan antarmuka pengguna yang mudah, mungkin kita perlu mempertimbangkan penggunaan sistem operasi *Windows* yang tersedia untuk itu (Nugroho, 2010).

### 2) Sistem Operasi Jaringan

Pilihan atas sistem operasi jaringan biasanya bergantung pada bagaimana informasi disampaikan pada sistem informasi yang kita kembangkan. Apabila sistem kita terorganisasi dengan baik dan jalur komunikasi cukup jelas, maka sistem operasi harus dapat memodelkan aliran informasi dalam sistem nyata yang ada. Misalkan jika sistem *client/server* seperti *Netware*, *Windows NT*, atau *Windows 2000* mungkin adalah yang paling sesuai. Apabila yang terjadi adalah sebaliknya, kita lebih mementingkan perpindahan ulang-alik antar pengguna, yaitu pada sistem *peer-to-peer*, maka sistem operasi seperti

Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

*Windows 3.1, Windows for Workgroups, Windows 95, atau Windows 98* mungkin adalah sistem operasi yang cocok. (Nugroho, 2010)

### 2.2.3 Pembangunan Sistem Informasi

Merupakan pembuatan sistem informasi yang baru atau pemodifikasian sistem yang ada. Pada dasarnya pengembangan sistem adalah berhubungan dengan proses analisis dan desain sistem.

Analisis Sistem adalah suatu proses untuk memahami sistem yang ada, termasuk mendiagnosa atau mengidentifikasi masalah dan memberikan alternatif solusi penyelesaiannya.

Desain Sistem adalah suatu proses pemecahan masalah dan proses kreativitas yang bertujuan membuat sistem baru yang dapat memenuhi tujuan utama sistem.

### 2.3 Android

*Android* adalah sistem operasi berbasis *kernel linux*. *Google* mengibaratkan *Android* sebagai tumpukan *software* dimana setiap tumpukan berisi program yang mendukung fungsi spesifik dari sistem operasi. Adapun susunan lapisan tersebut dari bawah ke atas adalah sebagai berikut: (Athoillah dkk, 2013)

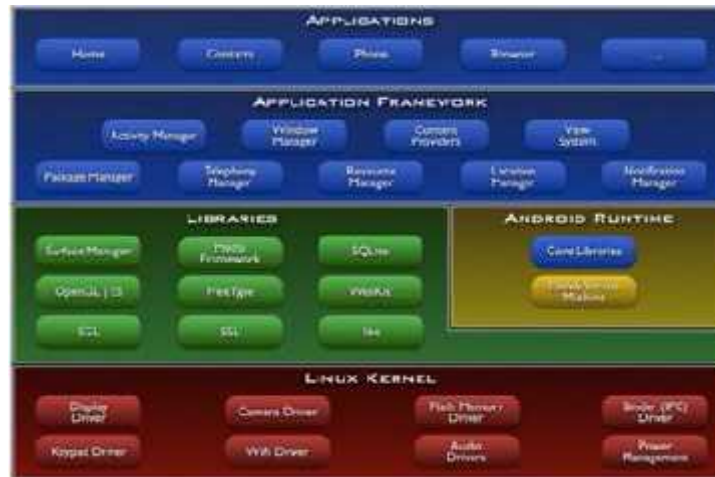
1. *Linux* sebagai *kernel*.
2. *Android runtime* dan *libraries* berisi *Dalvik Virtual Machine* dan kode-kode librari dalam bahasa C/C++
3. *Aplication framework* berisi program untuk mengatur fungsi-fungsi dasar *smartphone*
4. *Application*.



Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Adapun lapisan sistem operasi android terdapat pada Gambar 2.1 dibawah ini:



Gambar 2.1 Lapisan Sistem Operasi Android

(Sumber: Athoillah dkk, 2013)

Android diciptakan oleh sebuah perusahaan kecil bernama *Android Inc* pada tahun 2000, yang kemudian perusahaan tersebut dibeli oleh *Google Inc*, untuk mengembangkan android lebih lanjut, dibentuklah *Open Handset Alliance* (OHA) yang terdiri dari 34 perusahaan *software*, *hardware* dan telekomunikasi diantaranya yaitu *Google*, *HTC*, *Intel*, *Motorola*, *T-Mobile* dll (Athoillah dkk, 2013).

Sampai saat ini *Android* telah banyak berkembang hingga beberapa versi. *Android* versi 1.1 adalah versi pertama yang dirilis pada tahun 2009, kemudian berturut – turut muncul versi yang lain yang merupakan perbaikan demi perbaikan dari versi yang sebelumnya diantaranya ialah versi 1.5 (*Cupcake*), versi 1.6 (*Donut*), versi 2.0/2.1 (*Eclair*), versi 2.2 (*Froyo*), versi 2.3 (*Gingerbread*), versi 3.0/3.1 (*Honeycomb*), versi 4.0 (*ICS*), hingga yang terbaru saat ini adalah versi 4.1 (*Jellybean*) (Athoillah dkk, 2013).

## 2.4 Konsep Dasar Persediaan Barang

Penggunaan teknologi informasi untuk kegiatan mengolah data pada berbagai aktivitas perusahaan khususnya pada bagian gudang merupakan bagian penting bagi setiap perusahaan, dengan teknologi ini data yang telah diolah dapat menjadi sebuah informasi yang cepat dan tepat (Athoillah dkk, 2013).



#### Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

### 2.4.1 Defenisi Persediaan Barang

Persediaan barang menurut Ristono (2009) yang dikutip dari jurnal sistem informasi oleh Wahyu (2014) dapat diartikan sebagai barang-barang yang disimpan untuk digunakan atau dijual pada masa atau periode yang akan datang. Persediaan terdiri dari persediaan bahan baku, persediaan barang setengah jadi dan persediaan barang jadi. Persediaan bahan baku dan bahan setengah jadi disimpan sebelum digunakan atau dimasukkan dalam proses produksi, sedangkan persediaan barang jadi atau barang dagangan disimpan sebelum dijual atau dipasarkan. Dengan demikian, setiap perusahaan yang melakukan kegiatan usaha umumnya memiliki persediaan.

Persediaan dalam perusahaan, pengertian atau prosesnya tergantung dari jenis perusahaan tersebut. Jika perusahaan termasuk dalam kelompok perusahaan manufaktur berarti persediaan yang akan dikelola meliputi persediaan produk jadi, persediaan produk dalam proses, persediaan bahan baku, persediaan bahan penolong dan lainnya. Sedangkan jika perusahaan termasuk dalam kelompok perusahaan dagang, maka persediaan yang dikelola hanya satu macam saja yaitu persediaan barang dagang yang merupakan barang yang dibeli kemudian dijual kembali. Dari pengertian tersebut, maka dapat disimpulkan bahwa pengelolaan persediaan tergantung dari jenis perusahaannya (Munawaroh, 2006).

### 2.4.2 Prosedur Persediaan

Prosedur adalah suatu urutan kegiatan klerikal biasanya melibatkan beberapa orang dalam satu departemen atau lebih yang dibuat untuk menjamin penanganan secara seragam transaksi perusahaan yang terjadi secara berulang-ulang (Teuku, 2008).

Dari definisi tersebut dapat diambil kesimpulan bahwa suatu sistem terdiri dari jaringan prosedur sedangkan prosedur merupakan urutan kegiatan klerikal (Teuku, 2008).

Kegiatan klerikal (*klерical operation*) terdiri dari kegiatan berikut ini yang dilakukan untuk mencatat informasi: (Teuku, 2008)

Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

1. Mencatat
2. Menggandakan
3. Menghitung
4. Mengajukan
5. Memilih *Suplier*
6. Mendistribusi
7. Membandingkan
8. Laporan

Ada dua metode pencatatan persediaan, yaitu: (Teuku, 2008)

1. Metode mutasi persediaan (*perpectual inventory method*)

Setiap mutasi persediaan dicatat dalam kartu persediaan. Metode ini cocok digunakan dalam penentuan biaya bahan baku perusahaan yang harga pokoknya dikumpulkan dengan metode harga pokok pesanan.

2. Metode persediaan fisik (*physical inventory method*)

Hanya tambahan persediaan yang dicatat, sedangkan mutasi berkurangnya persediaan pemakaian tidak dicatat dalam kartu persediaan untuk mengetahui berapa harga pokok persediaan yang dipakai atau dijual harus dilakukan dengan perhitungan fisik sisa persediaan yang masih ada digudang pada akhir periode. Harga pokok persediaan awal periode ditambah dengan harga pokok persediaan pada akhir periode merupakan harga pokok persediaan yang dipakai selama periode akuntansi yang bersangkutan.

## 2.5 Metode Pendekatan Analisa Sistem Informasi

Dalam menganalisis suatu sistem informasi dapat dibantu dengan beberapa pendekatan metodologi. Salah satunya adalah pendekatan berorientasikan objek atau yang biasa disebut OOAD (*Object Oriented Analysis Design*).

### 2.5.1 Metode Object Oriented Analysis Design (OOAD)

Menurut Nugroho (2005) OOA adalah tahapan perangkat lunak dengan menentukan spesifikasi sistem (sering orang menyebutnya sebagai SRS atau





#### Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

*System Requirement Spesification*) dan mengidentifikasi kelas-kelas serta hubungannya satu terhadap yang lainnya. Untuk memahami spesifikasi sistem, kita perlu mengidentifikasi para pengguna atau yang sering disebut sebagai aktor-aktor. Siapa aktor-aktor yang akan menggunakan sistem dan bagaimana mereka menggunakan sistem.

Mencari objek-objek fisik pada sistem juga memungkinkan kita untuk mendapatkan informasi lebih lengkap objek-objek pada sistem yang bersangkutan. Objek-objek dapat bersifat mandiri, organisasi-organisasi, satuan informasi, gambar-gambar, atau apapun yang menyusun suatu aplikasi dalam konteks representasi dunia nyata dalam sistem yang sedang dikembangkan (Nugroho, 2005).

Adapun aktifitas utama dari OOAD menurut Nugroho (2005) adalah:

1. Menganalisis masalah domain
2. Menjelaskan sistem proses
3. Mengidentifikasi objek
4. Menentukan *atribut*
5. Mengidentifikasi operasi
6. Komunikasi objek

### 2.5.2 *Unified Modelling Language (UML)*

UML (*Unified Modeling Language*) adalah sebuah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Salahuddin, 2013). UML tidak menentukan metode untuk sistem-sistem pengembang, hanya catatan yang saat ini telah diterima luas sebagai standar untuk pemodelan objek (Whitten, 2004).

Sedangkan menurut Azis (2005) UML adalah sekumpulan simbol dan diagram untuk memodelkan *software*. Dengan menggunakan UML, desain *software* dapat diwujudkan dalam bentuk simbol dan diagram. Desain dalam bentuk simbol dan diagram, kemudian dapat diterjemahkan menjadi kode program.



Dengan menggunakan UML kita dapat membuat 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 ia lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek (Sumadyo, 2009).

Pada saat menggunakan UML untuk membangun perangkat lunak, pemodelan bisnis dapat membantu untuk memahami konteks sistem yang akan dibangun. Kedengarannya sepele, tetapi mempunyai konsekuensi yang serius pada kesuksesan atau kegagalan dalam merancang perangkat lunak. Jika gagal memahami proses bisnis, maka akan didapatkan penafsiran yang salah tentang apa yang perangkat lunak perlukan (Sholiq, 2006).

Tujuan utama dalam desain UML adalah (Haviluddin, 2011):

1. Menyediakan bagi pengguna (analisis dan desain sistem) suatu bahasa pemodelan visual yang ekspresif sehingga mereka dapat mengembangkan dan melakukan pertukaran model data yang bermakna.
2. Menyediakan mekanisme yang spesialisasi untuk memperluas konsep inti.
3. Karena merupakan bahasa pemodelan visual dalam proses pembangunannya maka UML bersifat independen terhadap bahasa pemrograman tertentu.
4. Memberikan dasar formal untuk pemahaman bahasa pemodelan.
5. Mendorong pertumbuhan pasar terhadap penggunaan alat desain sistem yang berorientasi objek (OO).
6. Mendukung konsep pembangunan tingkat yang lebih tinggi seperti kolaborasi, kerangka, pola dan komponen terhadap suatu sistem.
7. Memiliki integrasi praktik terbaik.

### 2.5.2.1 Use Case Diagram

*Use Case* atau diagram *Use Case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

Hak Cipta Dilindungi Undang-Undang

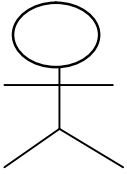
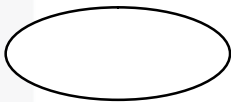


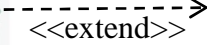
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

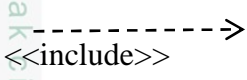
Secara kasar, *Use Case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Salahuddin, 2013). *Use Case* mendeskripsikan siapa yang akan menggunakan sistem dan dalam cara apa pengguna mengharapkan interaksi dengan sistem itu. (Whitten, 2004). Simbol diagram *Use Case* dapat dilihat pada Tabel 2.1 dibawah ini:

Tabel 2.1 Simbol Diagram *Use Case*

Simbol	Nama	Keterangan
	<i>Actor</i>	<i>Actor</i> adalah segala sesuatu yang berinteraksi langsung dengan sistem aplikasi komputer, seperti orang, benda atau lainnya. Tugas <i>actor</i> adalah memberikan informasi kepada sistem dan dapat memerintahkan sistem agar melakukan sesuatu tugas.
	<i>Use Case</i>	<i>Use Case</i> digambarkan sebagai lingkaran elips dengan nama <i>Use Case</i> dituliskan didalam <i>elips</i> tersebut.
	<i>Association Relationship</i>	Asosiasi digunakan untuk menghubungkan <i>actor</i> dengan <i>Use Case</i> .
	<i>Generalization Relationship</i>	<i>Generalization</i> menunjukkan hubungan antara elemen yang lebih umum ke elemen yang lebih spesifik.
	<i>Extend Relationship</i>	<i>Extend</i> menunjukkan bahwa suatu bagian dari elemen di garis tanpa panah bisa disisipkan kedalam elemen yang ada di garis dengan panah.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



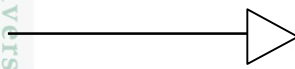
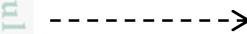
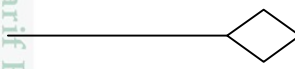
	<p><i>Include Relationship</i></p>	<p><i>Include</i> menunjukkan suatu bagian dari elemen (yang ada digaris tanpa panah) memicu eksekusi bagian dari elemen lain (yang ada di garis dengan panah)</p>
---	------------------------------------	--

### 2.5.2.2 Class Diagram

Diagram kelas atau *Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas, sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas (Salahuddin, 2013).

Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak *sinkron* (Salahuddin, 2013). Simbol diagram *Class* dapat dilihat pada Tabel 2.2 dibawah ini:




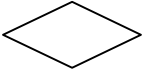

Tabel 2.2 Simbol Diagram *Class*

Simbol	Nama	Keterangan
	<i>Association</i>	Menghubungkan antara objek satu dengan objek lainnya.
	<i>Directed association</i> (asosiasi berarah )	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain.
	Generalisasi	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus).
	<i>Dependency</i> (Kebergantungan)	Relasi antar kelas dengan makna kebergantungan antar kelas.
	<i>Aggregation</i> (Agregasi)	Relasi antar kelas dengan makna semua-bagian ( <i>whole-part</i> ).

### 2.5.2.3 Activity Diagram

Menurut Salahuddin (2013) diagram aktivitas atau *Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Simbol diagram *Activity* dapat dilihat pada Tabel 2.1 dibawah ini:

Tabel 2.3 Simbol Diagram *Activity*

Simbol	Nama	Keterangan
	<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi.
	<i>Start State</i>	Bagaimana objek dibentuk atau diawali.
	<i>End State</i>	Bagaimana objek dibentuk dan dihancurkan.
	<i>Process</i>	Pilihan untuk mengambil keputusan.
	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.

### 2.6 Metode Pengembangan *Waterfall*

Merupakan kerangka berpikir atau cara yang ditempuh dalam menerapkan pengembangan sistem.

Dalam penelitian ini, model pengembangan sistem yang digunakan adalah *Waterfall*. *Waterfall* disebut model sekuensial linier atau alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari perencanaan, analisis, perancangan, implementasi, pengujian, dan pemeliharaan (Nugroho, 2010).



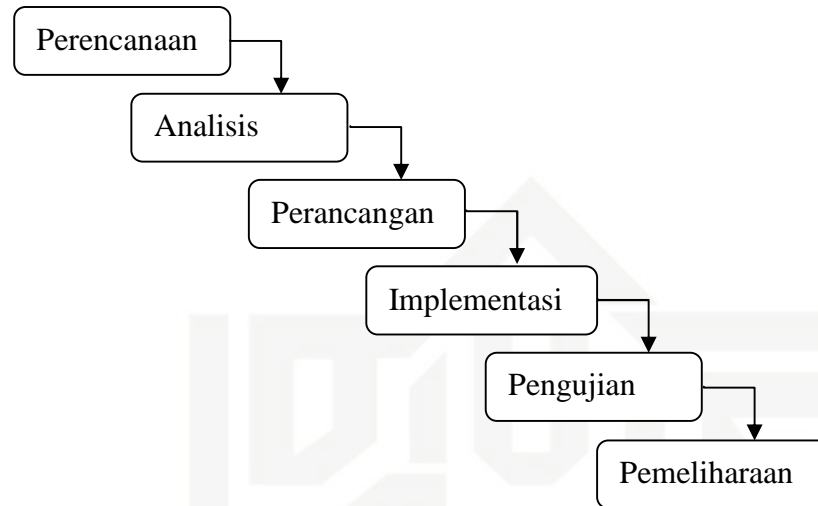
Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Adapun alur metode *Waterfall* dapat dilihat pada Gambar 2.2 dibawah ini:



Gambar 2.2 Ilustrasi Model *Waterfall*

(Sumber: Nugroho, 2010)

1. Tahap Perencanaan

Tahap perencanaan (*planning*) adalah menyangkut studi tentang kebutuhan pengguna (*user specification*), studi-studi kelayakan (*feasibility study*) baik secara teknis maupun secara teknologi serta penjadwalan pengembangan suatu proyek sistem informasi atau perangkat lunak.

2. Analisis

Langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian, wawancara atau studi literatur. Sistem analis akan menggali informasi sebanyak-banyaknya dari *user* sehingga akan tercipta sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh *user* tersebut. Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan sistem.

3. Perancangan

Tahapan dimana dilakukan penuangan pikiran dan perancangan sistem terhadap solusi dari permasalahan yang ada dengan menggunakan perangkat



#### Hak Cipta Diindungi Undang-Undang

1. Diarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Diarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

pemodelan sistem seperti UML diantara seperti *Class Diagram*, *Use Case Diagram*, *Activity Diagram* dan *Sequence Diagram*.

#### 4. Implementasi

Tahap implementasi adalah tahap dimana kita mengimplementasikan perancangan sistem ke situasi nyata, disini kita akan berurusan dengan pemilihan perangkat keras dan penyusunan perangkat lunak.

#### 5. Pengujian

Tahap pengujian adalah tahap dimana sistem yang baru diuji kemampuan dan keefektifannya sehingga didapatkan kekurangan dan kelemahan sistem yang kemudian dilakukan pengkajian ulang dan perbaikan terhadap aplikasi

#### 6. Pemeliharaan

Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan baru, atau karena pelanggan membutuhkan perkembangan fungsional.

### 2.7 Basis Data (*Data Base*)

Dalam buku Raharjo (2011) *database* didefinisikan sebagai kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil, dan dicari secara cepat. Sedangkan dalam (Fadlisyah, 2008) *database* adalah sekumpulan tabel-tabel yang saling berelasi, relasi tersebut bisa ditunjukkan dengan kunci dari tiap tabel yang ada. Satu Basis Data menunjukkan satu kumpulan data yang dipakai dalam satu lingkup perusahaan atau instansi.

Basis data (*database*) memiliki peran yang sangat penting dalam perusahaan. Informasi dapat diperoleh dengan cepat berkat data yang mendasarinya telah disimpan dalam basis data. Sebagai contoh, mekanisme pengambilan uang pada ATM (Anjungan Tunai Mandiri) sesungguhnya didasarkan pada pengambilan keputusan yang didasarkan pada basis data. Pertama, sistem akan memvalidasi keabsahan pemilik kartu dengan memeriksa *password* yang diberikan oleh orang tersebut. Dalam hal ini, *password* yang

diketik akan dicocokkan dengan *password* pada basis data. Jika sama, langkah berikutnya akan dilaksanakan, yaitu memeriksa saldo uang yang tercatat di basis data terhadap jumlah uang yang diambil. Jika memenuhi syarat, uang akan dikeluarkan oleh mesin (Kadir dkk, 2003).

## 2.8 Perangkat Lunak Pendukung Sistem

Perangkat lunak pendukung sistem merupakan alat bantu dalam pembangunan sistem sampai sistem diimplementasikan. Adapun *software* ataupun aplikasi-aplikasi pendukung sistem adalah sebagai berikut:

### 2.8.1 MySQL

*MySQL* adalah sebuah program *Database server* yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi user* serta menggunakan perintah standar *SQL (Structured Query Language)* (Nugroho, 2005). *MySQL* mempunyai dual-lisensi, yaitu *Open Source* di bawah *GPL (GNU General Public License)* dan *Commercial* dibawah *MySQL AB*. Ini berarti dalam satu produk *MySQL* yang sama tersedia dalam *GPL* yang bersifat gratis, dan yang bersifat komersi dari *MySQL AB* (Sanjaya, 2006).

Untuk membuat *database MySQL*, penelitian ini menggunakan aplikasi bawaaan *XAMPP*, yakni *Phpmyadmin*. *Phpmyadmin* adalah aplikasi *GUI (Graphical User Interface)* untuk *MySQL* yang berbasis *web*, sehingga kita bisa membuat, menghapus, dan melakukan perintah *MySQL* melalui tampilan *web*. *Xampp* yang bersifat *free* sehingga *xampp* dapat di *download* gratis pada alamat <https://www.apachefriends.org/download.html>.

Dalam buku Puspitosari (2011) disebutkan bahwa terdapat beberapa kelebihan yang dimiliki *MySQL*, diantaranya:

- a. *MySQL* dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah (*Multi user*).
- b. *MySQL* memiliki kecepatan yang bagus dalam menangani *query* sederhana, yaitu dapat memproses lebih banyak *SQL* per satuan waktu.

#### Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

- c. *MySQL* memiliki tipe kolom yang sangat kompleks, seperti *signed or unsigned integer, float, double, char, text, date, timestamp*, dan masih banyak lagi.
- d. *MySQL* memiliki operator dan fungsi secara penuh dan mendukung perintah *select* dan *where* dalam perintah *query*.
- e. *MySQL* memiliki keamanan yang bagus karena beberapa lapisan keamanan seperti level *subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang lengkap serta sandi terenkripsi.
- f. *MySQL* mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris.
- g. *MySQL* dapat melakukan koneksi dengan klien menggunakan protokol *TCP/IP, Unix socket (UNIX)*, atau *Named Pipes (NT)*.
- h. *MySQL* dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa.
- i. *MySQL* memiliki *interface* terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi *API (Application Programming Interface)*.

### 2.8.2 *Android Studio*

*Android Studio* adalah sebuah IDE untuk pengembangan aplikasi di platform *Android*. Sama seperti kombinasi antara *Eclipse* dan *Android Developer Tools (ADT)*, *Android Studio* juga dapat di-download di situs resmi *Android*: <http://developer.android.com/sdk/installing/studio.html>. Saat ini *Android Studio* memiliki versi terbaru yakni *Android Studio 2.2.2*.

Kenapa harus ada sebuah *IDE* baru? *SDK* sebelumnya di-*bundle* bersama dengan *Eclipse*, sementara *Android Studio* menggunakan *IntelliJ IDEA Community Edition*. Kedua *IDE* tersebut sama-sama memiliki penggemar ‘fanatik’-nya masing-masing. Beberapa pendukung *IntelliJ IDEA* sering mengatakan bahwa *Eclipse* terlalu rumit bagi pemula. Perbedaan lainnya? *Android Studio* menggunakan *Gradle* untuk manajemen proyeknya. Bagi yang belum pernah mendengar, *Gradle* adalah *build automation tool* yang dapat





1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

dikonfigurasi melalui *DSL* berbasis Groovy. Ini yang membedakan *Gradle* dari *Ant* atau *Maven* yang memakai *XML*. Penggunaan *DSL* berbasis *Groovy* menyebabkan *Gradle* lebih fleksibel dan dapat diprogram dengan mudah.

### 2.8.3 *Jquery*

*JQuery* adalah *library JavaScript multiplatform* yang dirancang untuk memudahkan penyusunan *client-side script* pada *file HTML*. Digunakan oleh 60 persen dari 10000 situs *web* paling banyak dikunjungi di dunia, *JQuery* adalah *library JavaScript* yang paling populer saat ini. *JQuery* merupakan sebuah perangkat lunak bebas sumber terbuka yang berada di bawah lisensi MIT.

*Sintaks JQuery* dirancang sedemikian rupa untuk memudahkan pengembang *website* dalam menavigasi dokumen, menyeleksi elemen-elemen DOM, menerapkan animasi, mengaplikasikan *events*, serta membangun aplikasi AJAX.

*JQuery* juga memampukan *developer* menciptakan berbagai *plugin* berbasis *library JavaScript*. Dengan *plugin-plugin* tersebut, pengembang situs *web* mampu menyusun sejumlah abstraksi untuk interaksi dan animasi sederhana, juga beberapa efek yang cukup kompleks dan berbagai widget yang dapat dikonfigurasi.

Karakter *library JavaScript* yang modular mendukung pengembangan laman web dinamis dengan berbagai fitur dan aplikasi berbasis *web (web app)*. Fitur-fitur inti *JQuery* meliputi penyeleksian, *traversal*, dan manipulasi elemen-elemen DOM dimampukan oleh sebuah selector *engine (Sizzle* mulai versi 1.3), telah mewujudkan suatu gaya pemrograman baru yang memadukan antara algoritma dan struktur data DOM. Gaya ini telah mempengaruhi arsitektur dari framework JavaScript lainnya seperti YUI v3 dan *Dojo*, dan di kemudian menstimulasi pengembangan Selectors API standar.

*Microsoft* dan *Nokia* membundel *JQuery* pada *platform* mereka. *Microsoft* memasukkannya dalam *Visual Studio* untuk digunakan dalam *framework ASP.NET AJAX* dan *ASP.NET MVC*, sedangkan *Nokia* mengintegrasikannya



#### Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

dalam platform pengembangan *widget Web Run-Time*. *JQuery* juga mulai dipakai pada MediaWiki sejak versi 1.16.

Situs resmi *JQuery* beralamat di *JQuery.com*. Pengembangannya dikelola oleh *JQuery Foundation* yang berada di *JQuery.org*. *Library* ini bisa diunduh di laman *JQuery.com/download*, atau diterapkan melalui sejumlah CDN termasuk *Google JQuery* CDN.

*JQuery*, pada intinya, adalah sebuah *library* yang berfungsi untuk memanipulasi *DOM*. *DOM* merupakan representasi struktural dari seluruh elemen pada sebuah laman web. Keberadaan *JQuery* menjadikan kerja pencarian, penyeleksian, dan manipulasi elemen-elemen *DOM* menjadi simpel dan mudah. Sebagai contoh, *JQuery* bisa dipergunakan untuk menemukan sebuah elemen dalam dokumen yang memiliki properti tertentu (misalnya: elemen dengan tag `h1`), kemudian mengubah satu atau beberapa atributnya (misalnya: warna, visibilitas), atau menjadikan elemen tersebut responsif terhadap suatu *event* (misalnya: klik *mouse*).

Selain penyeleksian dan manipulasi *DOM* dasar, *JQuery* menyajikan sebuah paradigma baru pada penanganan event oleh JavaScript. Penugasan event dan pendefinisian fungsi *event callback* dapat dilakukan dengan satu langkah dalam satu lokasi di dalam kode. *JQuery* juga dikembangkan untuk mendayagunakan berbagai fungsionalitas *JavaScript* yang paling sering dipakai (misalnya: fade in atau fade out ketika menentukan visibilitas elemen, juga bermacam animasi yang dimunculkan dengan memanipulasi properti-properti CSS).

Keuntungan dari pemanfaatan *JQuery* antara lain:

- a. Memisahkan *JavaScript* dan *HTML* Tanpa menggunakan atribut-atribut *HTML* untuk memanggil fungsi *Javascript* dalam penanganan event, *JQuery* bisa dipergunakan untuk menangani event dengan script JS saja.
- b. Singkat dan Jelas *JQuery* mengutamakan penulisan kode yang singkat dan jelas melalui berbagi fitur seperti fungsi-fungsi yang dapat dirangkaikan (*chain-able*) dan nama-nama fungsi yang pendek.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

c. Mengatasi masalah kompatibilitas antar-browser *JavaScript* engine pada berbagai browser memiliki perbedaan satu sama lain, sehingga script yang berjalan pada suatu *browser* bisa gagal pada browser lainnya. JQuery mengatasi segala inkonsistensi antar-*browser* tersebut dan menyajikan antarmuka yang konsisten bekerja pada semua *browser*.

d. Ekstensibel JQuery menjadikan pengembangan framework sangat simpel. Berbagai event, elemen, dan metode baru dapat dengan mudah ditambahkan dan digunakan ulang sebagai *plugin*.

JQuery memiliki fitur-fitur sebagai berikut:

- a. Penyeleksian elemen-elemen DOM menggunakan selector engine *Sizzle*. *Sizzle* merupakan perangkat lunak multiplatform sumber terbuka yang berawal sebagai proyek sampingan JQuery.
- b. Manipulasi DOM berdasarkan selektor CSS yang memanfaatkan nama-nama dan atribut elemen, misalnya *id* dan *class*, sebagai kriteria seleksi simpul-simpul DOM.
- c. Events
- d. Efek dan animasi
- e. AJAX
- f. Obyek-obyek *deferred* and *promise* untuk mengontrol pemrosesan asinkron
- g. Penguraian JSON
- h. Ekstensibilitas melalui plugin
- i. Sejumlah utilitas, misalnya: informasi agen pengguna (*user agent*), deteksi fitur perangkat
- j. Metode-metode kompatibilitas yang tersedia secara bawaan pada browser-browser modern, namun membutuhkan pencadangan pada browser yang lebih tua, misalnya ``inArray()`` dan ``each``
- k. Dukungan multi-browser

Baik versi 1.x maupun 2.x dari JQuery, keduanya mendukung “current-1 versions” (artinya versi stable terkini dari sebuah browser dan satu versi yang mendahuluinya) untuk *Firefox*, *Google Chrome*, *Safari*, dan *Opera*. Versi 1.x juga



mendukung IE 6 atau di atasnya. Sedangkan versi 2.x tidak lagi mendukung *Internet Explorer* 6 sampai 8 (yang mewakili kurang dari 28% dari seluruh browser yang digunakan) dan hanya mendukung IE 9 dan selanjutnya.

#### 2.8.4 *WebView*

Saat ini *Android Studio* dilengkapi dengan berbagai macam kemudahan dan kenyamanan dalam perancangan aplikasi berbasis *mobile*. Salah satu fitur di *Android Studio* adalah *Webview*. *Webview* sendiri merupakan sebuah *class* pada *android* yang menampilkan dan menjalankan aplikasi *mobile* berbasis *web*, dengan cara memanggil URL *web* yang akan diintegrasikan ke aplikasi *mobile*.

#### 2.8.5 *CSS Framework*

*CSS framework* adalah *library CSS* yang digunakan untuk membuat standarisasi *layout* dan *thema* dari *website*. Dengan *CSS framework* kita bisa membuat *Front-end website* dengan cepat mengingat *layout* dan tampilan sudah disediakan. Menurut Wikipedia, *CSS framework* adalah sebuah kumpulan *library* yang memudahkan dalam penggunaan dan pengolahan bahasa *CSS* atau *Cascading Stylesheet*.

#### 2.8.6 *Bootstrap*

*Bootstrap* digambarkan sebagai *CSS* sederhana namun dibangun dengan *pre-processor* yang menyediakan lebih banyak daya dan fleksibilitas ketimbang *CSS* standar. Awalnya dirilis pada Jumat, 19 Agustus, 2011, dan sudah memiliki lebih dari 20 produk *release* termasuk yang terbesar adalah versi 2 dan versi 3. Hadirnya *Bootstrap 2*, maka fungsionalitas responsif terhadap seluruh kerangka sebagai *stylesheet opsional* telah di tambahkan dan pada *Bootstrap 3*, maka responsif secara *default* telah ditambahkan untuk pendekatan pertamanya terhadap ponsel.

*Bootstrap* merupakan sebuah *framework* yang dikembangkan oleh tim *Twitter* yang berfungsi sebagai kerangka kerja untuk memudahkan dan mempercepat dalam membuat tampilan *website* yang *responsive* dan terstandar. *Bootstrap* menyediakan komponen-komponen antarmuka siap pakai yang telah



Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

dirancang sedemikian rupa untuk digunakan dalam membuat tampilan halaman *website responsive*. *Bootstrap* juga menyediakan sarana untuk membuat layout halaman *web* dengan mudah dan rapi yaitu menggunakan *grid system* 12 kolom. *Bootstrap* dibangun dengan teknologi HTML, CSS dan *Javascript* yang dapat membuat *layout* halaman *website*, *table*, tombol, *form*, navigasi, dan komponen lainnya dalam sebuah *website* hanya dengan memanggil fungsi *class* dalam berkas HTML.

Berikut ini adalah beberapa keunggulan atau kelebihan menggunakan *WRD Bootstrap*:

- 1) Waktu pembuatan yang lebih cepat: Karena elemen-elemen yang biasa ada dalam sebuah *website* pada umumnya sudah dibuatkan *class*-nya oleh *Bootstrap* ini, jadi kita tinggal memanggilnya dalam *theme*.
- 2) *Template* yang menggunakan *Bootstrap* lebih rapi : Mungkin ini akan sedikit relatif, karena kebiasaan masing-masing *developer* membuat *coding* pasti berbeda-beda. Namun pada dasarnya, karena *bootstrap* sudah terstruktur, untuk modifikasi *class* dan penambahan *class* memang sebaiknya mempertahankan kerapihan kode yang sudah ada lebih dulu.
- 3) *Template* yang menggunakan *Bootstrap* lebih ringan.

### 2.8.7 Highchart

*Highchart* adalah *charting library* yang ditulis dengan bahasa *Javascript* murni. *Highchart* menawarkan *intuitif*, grafik interaktif ke situs *web* atau aplikasi *web*. Saat ini *highchart* mendukung grafik tipe *line*, *spline*, *area*, *areaspline*, *column*, *bar*, *pie*, dan *scatter*. *Highchart* sudah dapat berjalan di semua *browser* modern termasuk pada *iPhone/iPad*. *Browser* standar menggunakan SVG untuk proses *render* grafik, sedangkan *internet explorer* menggunakan VML untuk penggambaran grafik.

### 2.8.8 Nox APP Player Emulator

Nox APP Player adalah salah satu emulator Android keluaran terbaru. Sama seperti emulator lainnya. Mengklaim sebagai salah satu emulator android

### 2.8.9 Notepad ++

Notepad++ adalah sebuah penyunting teks dan penyunting kode sumber yang berjalan di sistem operasi windows. Notepad++ menggunakan komponen *Scintilla* untuk dapat menampilkan dan menyuntingan teks dan berkas kode sumber berbagai bahasa pemrograman.

Notepad++ di distribusikan sebagai perangkat lunak bebas. Proyek ini dilayani oleh Sourceforge.net dengan telah diunduh lebih dari 27 juta kali dan dua kali memenangkan penghargaan *Source Forge Community Choice Award for Best Developer Tool*.

Terdapat beberapa bahasa pemrograman yang didukung oleh Notepad++. Didukung dalam hal ini adalah dimengerti dan diterjemahkan menjadi teks oleh Notepad++. Misalnya pada C++, fungsi-fungsinya akan di masukan kedalam daftar fungsi dan kata-katanya akan berubah warna sesuai dengan makna kata tersebut di C++.

Berikut sejumlah bahasa pemrograman yang didukung oleh Notepad++ sejak versi 5.9.3:

1. *ActionScript*, Ada, ASP, *Assembler*, *autoIt*
2. *Batch*
3. C, C++, C#, Caml, Cmake, COBOL, CSS
4. D, Diff
5. *Flash ActionScript*, Fortran

Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

6. Gui4CLI
7. Haskell, HTML
8. Berkas INI, InnoSetup
9. Java, Javascript, JSP
10. KiXtart
11. LISP, Lua
12. Makefile, Matlab, MS-DOS
13. NSIS
14. Objective-C
15. Pascal, Perl, PHP, *Postscript*, *PowerShell*, *Properties file*, Python
16. R, *Resource file*, Ruby
17. Shell, Scheme, Smalltalk, SQL
18. TCL, TeX
19. XML
20. YAML

#### 2.8.10 *MyPublicWifi*

*MyPublicWifi* adalah sebuah aplikasi yang digunakan untuk membuat jaringan dimana laptop atau komputer kita sebagai akses poinnya. Siapapun bisa menggunakan internet yang terhubung dengan *wifi* yang dibagikan. *MyPublicWifi* juga dilengkapi dengan dukungan *firewall* yang dapat digunakan untuk membatasi akses pengguna ke server tertentu. Selain itu, *MyPublicWifi* juga dapat merekam dan melacak semua halaman URL yang dikunjungi pada *virtual wifi* yang dibuat.

#### 2.8.11 *UCBrowser*

UCWeb *Inc.* (UCWeb), sebuah perusahaan Alibaba *Group*, adalah penyedia teknologi dan layanan perangkat lunak *internet mobile* terkemuka. Sejak awal tahun 2004, misi UCWeb yaitu memberikan pengalaman *mobile internet* yang lebih baik untuk miliaran pengguna di seluruh dunia. portofolio produk internasional termasuk layanan *mobile browsing service* (*UC Browser*), *app and*



#### Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

*game distribution platforms (9Apps and 9Game), a mobile traffic platform (UC Union) dan sebagainya.*

UC *Browser* merupakan produk unggulan dari UCWeb, tersedia lebih dari 3.000 model yang berbeda dari lebih dari 200 produsen ponsel. Selain itu, UC *Browser* kompatibel dengan semua sistem operasi utama seperti *Android, iOS, Windows Phone, Symbian, Win CE, Jawa, MTK, dan Blackberry*. Pada bulan Juni 2011, UCWeb merilis mesin U3 sendiri. *Browser* berbasis kernel U3 memberikan pengguna pengalaman *web surfing* lebih cepat, lebih nyaman dan lebih aman. Setelah menjadi *browser* ponsel No.1 di Cina dan India, UC *Browser* telah menjadi perngat ketiga *browser* ponsel terbesar di dunia pada akhir 2014 sesuai *StatCounter* dalam hal pangsa pasar *pageview* per bulan. Pada akhir Maret 2015, UC *Browser* telah menjadi *browser* ponsel No.1 di Indonesia menurut *StatCounter*.

Hari ini, UCWeb terus berkembang dengan cepat, dengan 80% dari 3.000 karyawan yang terlibat dalam penelitian dan pengembangan produk. UCWeb memiliki lebih dari 200 paten, baik yang tertunda dan diterima, di bidang *mobile browsing*.

UCWeb telah mendirikan kantor lokal dan tim di Cina, India, Vietnam, Rusia dan Indonesia, melayani pengguna di lebih dari 150 negara dan wilayah di seluruh dunia. Sebagai langkah perusahaan ke arah globalisas, UCWeb akan terus memberikan pengalaman *mobile internet* yang lebih baik untuk miliaran pengguna di seluruh dunia.