

BAB II

TINJAUAN PUSTAKA

2.1 Sistem Informasi

Sistem Informasi adalah suatu kombinasi yang terorganisasi dari orang, perangkat keras, perangkat lunak, jaringan komunikasi, sumber data, kebijakan dan prosedur yang menyimpan, mengambil, mengubah, dan menyebarkan informasi dalam suatu organisasi (O'Brien & Marakas, 2010).

2.2 *Customer Relationship Management (CRM)*

Customer Relationship Management adalah pendekatan strategi manajemen dalam upaya menciptakan, mengembangkan dan mewujudkan hubungan yang saling menguntungkan dengan pelanggan dalam jangka panjang, khususnya terhadap pelanggan potensial, dalam upaya memaksimalkan *customer value* (nilai pelanggan) dan *corporate profitability* (Buttle & Maklan, 2015).

Ada empat fase dalam CRM (Ngai dkk, 2009) yaitu :

1. Identifikasi konsumen (*customer identification*).
2. Membangun daya tarik terhadap konsumen (*customer attraction*).
3. Mempertahankan konsumen yang ada (*customer retention*).
4. Mengembangkan konsumen / pelanggan (*customer development*).

Sedangkan Menurut (Park & Kim, 2003) ada tiga fase dalam CRM yang terdiri dari :

1. *Customer Acquisition*
2. *Customer Retention*
3. *Customer Expansion (Development)*

Penelitian ini berfokus pada fase ke 3 dari (Park & Kim, 2003) yakni *Customer Development*.

2.2.1 Pengembangan Pelanggan (*Customer Development*)

Pengembangan pelanggan adalah proses menumbuhkan nilai mempertahankan pelanggan. Perusahaan umumnya berusaha menjual silang dan menjual produk ke basis pelanggan sambil tetap memperhatikan kepuasan

pelanggan. *Cross-selling* adalah menjual produk dan layanan tambahan kepada pelanggan lama. Sedangkan *Up-selling* dapat didefinisikan sebagai menjual produk dan layanan dengan harga lebih tinggi atau lebih tinggi ke pelanggan yang sudah ada (Buttle & Maklan, 2015).

Menurut (Buttle dan Maklan, 2015) Pelanggan umumnya tidak menanggapi secara positif upaya terus-menerus dan berulang untuk menjual produk dan layanan tambahan yang tidak terkait dengan kebutuhan mereka. Memang, ada argumen bahwa perusahaan harus berusaha menjual ke bawah bila sesuai. Ini berarti mengidentifikasi dan memberikan solusi biaya yang lebih rendah untuk masalah pelanggan, meskipun itu berarti membuat margin lebih rendah. Pelanggan mungkin menganggap up-selling sebagai oportunistik dan eksploitatif, sehingga mengurangi tingkat kepercayaan yang mereka miliki di pemasok, dan menempatkan hubungan pada risiko. Namun, kepemilikan multi-produk menciptakan ikatan struktural yang mengurangi risiko pembubaran hubungan. Ada sejumlah teknologi CRM yang berguna untuk tujuan pengembangan pelanggan terdiri dari:

1. Data Mining

Pemasaran *cross sell* dan *up sell* didasarkan pada data mining. Riwayat transaksional mencatat apa yang telah dibeli pelanggan. Data mining dapat memberi tahu anda kemungkinan pelanggan membeli produk lain (kecenderungan untuk membeli), berdasarkan riwayat atau profil transaksi mereka. Secara *online* dan telepon bank pertama kali menggunakan kecenderungan untuk membeli *scores* untuk menjalankan pemasaran *cross-selling* yang ditargetkan melalui surat langsung dan *call center*. Mereka menargetkan tingkat konversi yang tinggi dengan menindaklanjuti panggilan telepon keluar yang terkait.

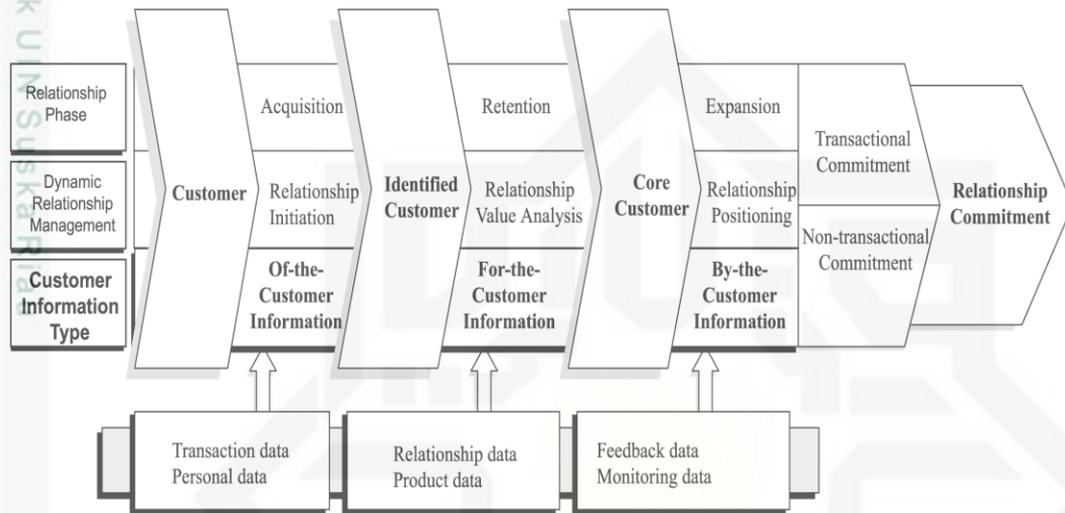
2. Customization

Penawaran *cross sell* dan *up sell* dapat disesuaikan pada segmen atau tingkat pelanggan unik, berdasarkan pada riwayat dan profil transaksional dari target. Juga personalisasi adalah komunikasi kepada pelanggan dan saluran komunikasi seperti email, media sosial, SMS atau panggilan telepon.

3. *Integrated Customer Communications*

Praktisi CRM umumnya lebih suka bahwa pesan yang disampaikan kepada pelanggan konsisten disemua saluran.

Pada jurnal (Park & Kim, 2003), dia menggunakan bahasa pengembangan pelanggan ini untuk *expansion*. Hal ini dapat dilihat pada Gambar 2.1.



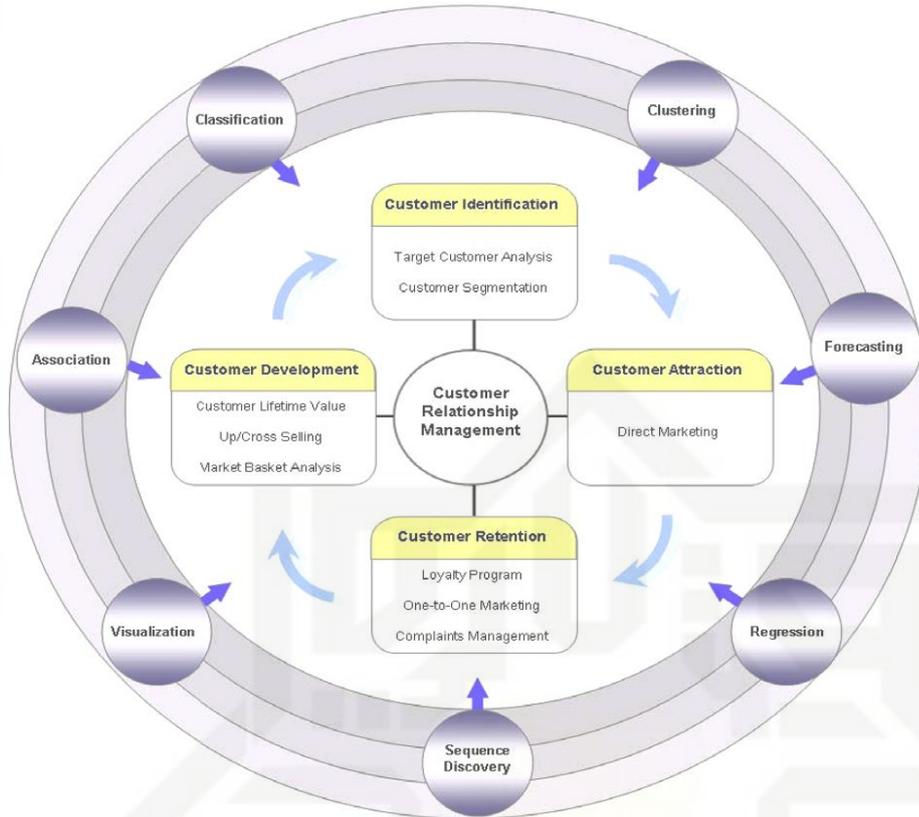
Gambar 2.1 *Framework of Dynamic CRM*

(Sumber: Park dan Kim, 2003)

Pada Gambar 2.1 di atas menunjukkan kerangka kerja *Dynamic CRM*. Dalam kerangka kerja tersebut terdapat beberapa fase yaitu *Relationship Phase*, *Dynamic Relationship Management*, dan *Customer Information Type*. Dalam *Relationship Phase* terdiri dari *Acquisition*, *Retention* dan *Expansion*. Sedangkan dalam jurnal Ngai (2009) menggunakan *development*. Secara grafis, klasifikasi framework pada teknik data mining dalam CRM diusulkan dan ditunjukkan pada Gambar 2.2 berikut.

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.



Gambar 2.2 Klasifikasi Framework untuk teknik data mining dalam CRM

(Sumber: Ngai dkk, 2009)

Hal tersebut berdasarkan pada penjelasan dari literatur pada data mining dalam CRM membantu untuk mengidentifikasi dimensi CRM utama dan teknik data mining untuk aplikasi dari teknik data mining dalam CRM. Dalam konteks dari CRM, data mining dapat dilihat sebagai proses gerakan bisnis yang terarah pada penemuan dan tetap menggunakan pengetahuan yang menguntungkan dari data organisasi (Ling, 2001). Hal tersebut dapat digunakan untuk panduan dalam membuat keputusan dan peramalan dari efek dan keputusan. Sebagai contoh, data mining dapat meningkatkan kecepatan respon dari kampanye pemasaran dengan segmentasi *customer* ke dalam kelompok-kelompok dengan karakteristik dan kebutuhan yang berbeda-beda. Itu dapat memprediksi bagaimana mungkin seorang *customer* yang ada untuk membawa bisnisnya kepada kompetitor (Carrier dan Povel, 2003) . Setiap elemen dari CRM dapat didukung oleh model data mining, yang secara umum termasuk asosiasi, klasifikasi, *clustering*, peramalan, regresi, urutan penemuan dan visualisasi.

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.

1. Asosiasi

Asosiasi bertujuan untuk menetapkan hubungan antara item-item yang ada bersama-sama dalam sebuah catatan yang diberikan (Ahmed, 2004). Analisis market basket dan program lintas penjualan adalah contoh tipe untuk model asosiasi yang biasanya dipakai. *Tools* umum untuk model asosiasi adalah statistika dan algoritma apriori (Ngai dkk, 2009).

2. Klasifikasi

Klasifikasi adalah salah satu dari model pembelajaran yang umum dalam data mining. Klasifikasi mengarah pada membangun sebuah model untuk memprediksi perilaku customer di masa yang akan datang melalui klasifikasi baris database ke dalam sebuah nomor dari kelas yang sudah dikenal berdasarkan pada kriteria tertentu (Ahmed, 2004). *Tools* umum untuk klasifikasi adalah *neural networks*, *decision trees*, dan *if-then-else rules*.

3. Clustering

Clustering adalah tugas dari segmentasi populasi/kelompok yang berbeda ke dalam satu atau lebih dari *cluster* yang sama (Ahmed, 2004). Ini berbeda dengan klasifikasi yang *cluster-cluster*nya adalah tidak diketahui pada saat algoritma dimulai. Dengan kata lain, tidak ada *cluster* yang sudah dikenal. *Tools* umum untuk *clustering* termasuk *neural networks* dan analisis diskriminasi (Ngai dkk, 2009).

4. Peramalan

Peramalan memperkirakan nilai masa depan berdasarkan pada pola *record*. Hal tersebut dilakukan dengan nilai keluar-masuk secara terus menerus (Ahmed, 2004). Ini berhubungan dengan hubungan model dan logis dari model pada waktu yang sama di masa yang akan datang. *Tools* umum untuk peramalan termasuk *neural networks* dan analisis kelangsungan hidup (Ngai dkk, 2009).

5. Regresi

Regresi adalah jenis teknik estimasi secara statistik menggunakan pemetaan tiap data objek menjadi sebuah nilai *real* yang menyediakan nilai prediksi (Carrier dan Povel, 2003). Penggunaan regresi termasuk *curve fitting*,

prediksi (termasuk peramalan), model dari hubungan secara kebetulan, dan pengujian hipotesa secara ilmiah tentang hubungan antara variabel-variabel.

Tools umum untuk regresi termasuk regresi linear dan regresi logistik.

6. Urutan penemuan

Urutan penemuan adalah identifikasi dari asosiasi atau pola lebih dari (Carrier dan Povel, 2003). Tujuannya adalah untuk model keadaan dari proses membuat urutan atau untuk mengekstrak dan laporan deviasi dan trend yang lebih dari waktu (Mitra, 2002). *Tools* umum untuk urutan penemuan adalah statistika dan *set theory*.

7. Visualisasi

Visualisasi menjadi presentasi dari data sehingga *users* dapat melihat pola yang kompleks (Shaw, 2001). Hal tersebut digunakan dalam kombinasi dengan model data mining yang lain untuk menyediakan pemahaman secara lebih jelas dari penemuan pola-pola atau hubungan. Contoh dari model visualisasi adalah grafik 3D, “Hygraphs” dan SeeNet” (Shaw, 2001).

Untuk penelitian ini berfokus pada jurnal (Ngai, 2009) karna dia menggunakan *customer development*.

2.3 Data Mining

Data mining adalah proses menemukan pola dan pengetahuan menarik dari data dalam jumlah besar. Sumber data dapat mencakup *database*, gudang data, *web*, repositori informasi lainnya atau data yang dialirkan ke sistem secara dinamis (Han dkk, 2002).

2.3.1 Proses Data Mining

Menurut Han dkk, (2002) data mining terdiri dari beberapa proses. Adapun proses dalam data mining yaitu:

1. *Data cleaning*

Proses *cleaning* mencakup antara lain membuang duplikasi data, memeriksa data yang tidak konsisten, dan memperbaiki kesalahan pada data seperti kesalahan cetak.

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. *Data integration*

Pada tahap data *integration* dilakukan pengecekan kombinasi data terhadap data yang berasal dari banyak sumber.

3. *Data selection*

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi. Data hasil seleksi yang akan digunakan untuk proses data mining disimpan dalam suatu berkas terpisah dari basis data operasional.

4. *Data transformation*

Merupakan proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses data mining. Proses ini merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data. Kadang-kadang transformasi data dan konsolidasi dilakukan sebelum proses seleksi data, khususnya dalam kasus data *warehousing*. Reduksi data juga dapat dilakukan untuk mendapatkan representasi yang lebih kecil dari data asli tanpa mengorbankan integritasnya.

5. *Data mining*

Pemilihan tujuan dari proses *Knowledge Discovery in Databases* (KDD) misalnya klasifikasi, regresi, *clustering*, dll. Proses data mining adalah proses mencari pola atau informasi menarik dalam data terpilih menggunakan teknik dan metode tertentu. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses *Knowledge Discovery in Databases* (KDD) secara keseluruhan.

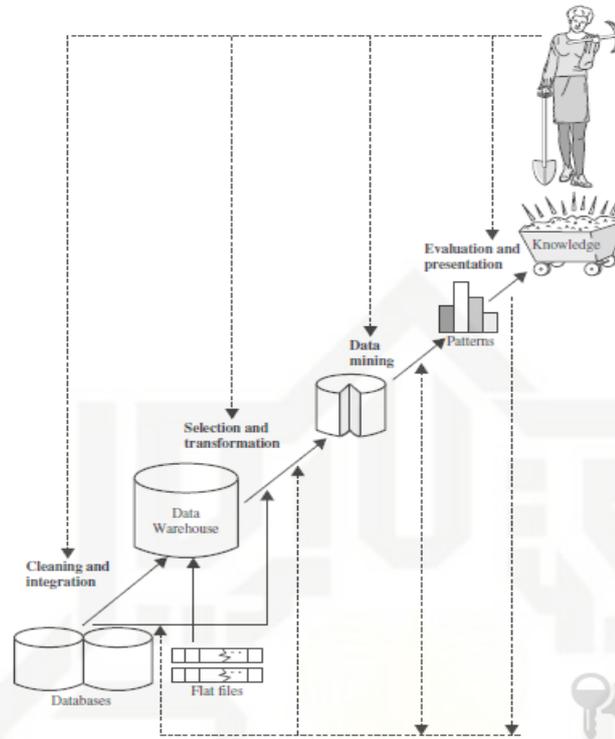
6. *Pattern evaluation*

Proses yang mengidentifikasi pola yang benar-benar menarik yang mewakili pengetahuan berdasarkan beberapa ukuran tindakan meliputi hipotesa sebelumnya.

7. *Knowledge presentation*

Pola informasi yang dihasilkan dari proses data mining perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan.

Untuk tahapan yang lebih jelas, dapat dilihat pada Gambar 2.3.



Gambar 2.3 Proses Data mining

(Sumber: Han dkk, 2002)

2.3.2 Pengelompokan *Data Mining*

Menurut Larose (2005), *data mining* dibagi menjadi beberapa kelompok berdasarkan tugas/pekerjaan yang dapat dilakukan, yaitu:

1. Deskripsi

Terkadang peneliti dan analisis secara sederhana ingin mencoba mencari cara untuk menggambarkan pola dan kecenderungan yang terdapat dalam data. Deskripsi dari pola kecenderungan sering memberikan kemungkinan penjelasan untuk suatu pola atau kecenderungan.

2. Estimasi

Estimasi hampir sama dengan klasifikasi, kecuali variabel target estimasi lebih kearah numerik dari pada ke arah kategori. Model dibangun menggunakan baris data (*record*) lengkap yang menyediakan nilai dari variabel target sebagai nilai prediksi. Selanjutnya, pada peninjauan

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.

berikutnya estimasi nilai dari variabel target dibuat berdasarkan nilai variabel prediksi.

3. Prediksi

Prediksi hampir sama dengan klasifikasi dan estimasi, kecuali bahwa dalam prediksi nilai dari hasil akan ada di masa mendatang. Beberapa metode dan teknik yang digunakan dalam klasifikasi dan estimasi dapat pula digunakan (untuk keadaan yang tepat) untuk prediksi.

4. Klasifikasi

Dalam klasifikasi, terdapat target variabel kategori. Sebagai contoh, penggolongan pendapatan dapat dipisahkan dalam tiga kategori, yaitu pendapatan tinggi, pendapatan sedang, dan pendapatan rendah.

5. Pengklasteran (*Clusterring*)

Pengklasteran merupakan pengelompokan *record*, pengamatan, atau memperhatikan dan membentuk kelas obyek-obyek yang memiliki kemiripan. Klaster adalah kumpulan *record* yang memiliki kemiripan satu dengan yang lainnya dan memiliki ketidakmiripan *record* dalam klaster yang lain. Berbeda dengan klasifikasi, pada pengklasteran tidak ada variabel target. Pengklasteran tidak melakukan klasifikasi, mengestimasi, atau memprediksi nilai dari variabel target, akan tetapi, algoritma pengklasteran mencoba untuk melakukan pembagian terhadap keseluruhan data menjadi kelompok-kelompok yang memiliki kemiripan (homogen), yang mana kemiripan *record* dalam satu kelompok akan bernilai maksimal, sedangkan kemiripan dengan *record* dalam kelompok lain akan bernilai minimal.

6. Asosiasi

Tugas asosiasi dalam *data mining* adalah untuk menemukan atribut yang muncul dalam satu waktu. Salah satu implementasi dari asosiasi adalah *market basket analysis* atau analisis keranjang belanja.

2.3.3 Algoritma Apriori

Algoritma Apriori adalah suatu algoritma dasar yang diusulkan oleh Agrawal & Srikant pada tahun 1994 untuk penentuan *frequent itemsets* untuk aturan asosiasi *boolean* (Han dkk, 2002). Algoritma apriori bertujuan untuk menemukan semua aturan apriori yang memenuhi syarat minimum *support* dan minimum *confidence* (Pane, 2013).

2.3.4 Market Basket Analysis

Market basket analysis adalah suatu metodologi untuk melakukan analisis *buying habit* konsumen dengan menemukan asosiasi antar beberapa *item* yang berbeda, yang diletakkan konsumen dalam *shopping basket* (keranjang belanja) yang dibeli pada suatu transaksi tertentu (Han dkk, 2002). Tujuan dari *market basket analysis* adalah untuk mengetahui produk-produk mana yang mungkin akan dibeli secara bersamaan (Han dkk, 2002).

2.3.5 Analisis Asosiasi

Analisis asosiasi atau *association rule mining* bertujuan untuk membangun hubungan antara item yang ada bersama dalam rekaman tertentu (Ahmed, 2004). Analisis keranjang pasar dan program penjualan silang adalah contoh tipikal yang digunakan untuk pemodelan asosiasi. *Tools* untuk pemodelan asosiasi adalah statistik dan algoritma apriori. Penelitian ini menggunakan metode *association* dengan algoritma apriori. *Interestingness measure* yang dapat digunakan dalam *data mining* adalah (Kusrini & Luthfi, 2009):

1. *Support*

Suatu ukuran yang menunjukkan seberapa besar tingkat dominasi suatu *item* atau *itemset* dari keseluruhan transaksi.

2. *Confidence*,

Suatu ukuran yang menunjukkan hubungan antar dua *item* secara *conditional* (berdasarkan suatu kondisi tertentu).

Metodologi dasar analisis asosiasi terbagi menjadi 2 tahap, yakni melakukan analisa pola frekuensi tinggi (*frequent pattern*) dan berikutnya adalah proses pembentukan aturan asosiasi (Kusrini & Luthfi, 2009).

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

Hak Cipta Dilindungi Undang-Undang

UIN SUSKA RIAU

Sultan Syarif Kasim Riau

2.3.6 Analisis Pola Frekuensi Tinggi

Nilai *support* sebuah *item* diperoleh dengan menggunakan rumus berikut:

$$Support\ A = \frac{\text{jumlah transaksi mengandung A}}{\text{total transaksi}} \quad (2.1)$$

Sementara, nilai *support* dari 2 *item* diperoleh dengan menggunakan rumus:

$$Support\ (A,B) = P(A \cap B) \quad (2.2)$$

$$Support = \frac{\epsilon \text{ transaksi mengandung A dan B}}{\epsilon \text{ transaksi}} \quad (2.3)$$

Frequent item set menunjukkan *item set* yang memiliki frekuensi kemunculan lebih dari nilai minimum yang ditentukan. (Kusrini & Luthfi, 2009).

2.3.7 Pembentukan Aturan Asosiasi (*Association Rules*)

Menurut (Sari & Novita, 2013), setelah semua pola frekuensi tinggi ditemukan, barulah dicari aturan asosiasi yang memenuhi syarat minimum untuk *confidence* dengan menghitung *confidence* aturan asosiatif $A \rightarrow B$. Nilai *confidence* dari aturan $A \rightarrow B$ diperoleh dengan rumus berikut:

$$Confidence = \frac{\epsilon \text{ transaksi mengandung A dan B}}{\epsilon \text{ transaksi mengandung A}} \quad (2.4)$$

Untuk menentukan aturan asosiasi yang akan dipilih maka harus diurutkan berdasarkan $Support \times Confidence$. Aturan diambil sebanyak n aturan yang memiliki hasil terbesar (Sari & Novita, 2013).

2.4 Penelitian terdahulu

Penelitian tentang *Customer Relationship Management* dan Algoritma Apriori pernah dilakukan oleh:

1. Maghfirah dkk, (2015) membahas *data mining* memiliki bermacam-macam teknik dan algoritma. Oleh karena itu, dari beberapa teknik dan algoritma dari *Data Mining* tersebut ada satu teknik dan algoritma yang kemungkinan merupakan yang paling tepat untuk digunakan dalam kasus analisis *Customer Relationship Management (CRM)* di industri perbankan. Untuk menentukan pilihan dari teknik data mining tersebut harus berdasarkan pada karakteristik data dan kebutuhan bisnis itu sendiri. Selain itu, harus mempertimbangkan juga elemen CRM apa yang akan

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.

difokuskan sehingga nantinya akan dapat dianalisis dengan teknik dan algoritma data mining yang tepat untuk sistem CRM tersebut dalam industri perbankan.

2. Tama (2010) *data mining* menjadi konsep yang berperan sangat penting untuk mengekstrak data transaksi menjadi sebuah pengetahuan (*knowledge*) yang didapatkan dari *association rules*. Penetapan *cross-selling* dengan tidak hanya diperuntukkan bagi kasus perusahaan ritel seperti supermarket, minimarket, pasar swalayan, ataupun *hypermarket*, namun berlaku pula bagi perusahaan ritel umum yang telah memiliki pelanggan tetap.
3. Wandu dkk, (2012) algoritma apriori dapat menghasilkan rekomendasi buku berdasarkan transaksi peminjaman buku yang ada. Apriori sangat bergantung pada banyak nya buku yang ada pada tiap transaksinya (*item set*). Dalam penelitian ini batasan minsup tidak bisa lebih dari empat, karena data transaksi terbanyak ada pada satu dan dua buku per transaksi.
4. Pane (2013) merek produk elektronik yang paling banyak terjual adalah Acer dan Toshiba, dengan diketahuinya produk yang paling banyak terjual tersebut, sehingga perusahaan dapat menyusun strategi pemasaran untuk memasarkan produk dengan merek lain dengan meneliti apa kelebihan produk yang paling banyak terjual tersebut dengan produk lainnya dan dapat menambah persediaan Acer dan Toshiba.
5. Gunadi & Sensuse (2012) *market basket analysis* dengan menggunakan algoritma Apriori dan *FP-growth* terhadap atribut- atribut penjualan produk buku telah menghasilkan sejumlah aturan asosiasi yang berbeda antara algoritma yang satu dengan yang lainnya. Analisa yang dilakukan terhadap tingkat kekuatan aturan-aturan asosiasi menunjukkan bahwa aturan-aturan asosiasi yang dihasilkan oleh algoritma Apriori memiliki tingkat kekuatan yang lebih tinggi dibandingkan dengan yang dihasilkan oleh algoritma *FP-growth*.

2.5 Profil PT. Herba Penawar Alwahida Indonesia

PT. Herba Penawar Alwahida Indonesia, yang kemudian dikenal sebagai HPAI, merupakan salah satu perusahaan Bisnis Halal *Network* di Indonesia yang fokus pada produk-produk herbal. HPAI sesuai dengan akta pendirian perusahaan, secara resmi didirikan pada tanggal 19 Maret 2012.

HPAI dibangun dari perjuangan panjang yang bertujuan menjayakan produk-produk halal dan berkualitas berazaskan Thibbunnabawi, serta dalam rangka membumikan, memajukan dan mengaktualisasikan ekonomi Islam di Indonesia melalui *entrepreneurship*.

2.5.1 Management HPAI

Dalam *management* HPAI memiliki dewan, yang terdiri dari dewan syariah, dewan komisaris dan dewan direksi. Adapun anggota dari masing-masing dewan, yaitu:

Adapun yang termasuk dalam Dewan Syariah adalah sebagai berikut:

1. Dr. H. Mawardi Muhammad Saleh, MA
2. Prof. Drs. H. M. Nahar Nahrawi, SH, MM (BPH DSN-MUI)
3. Dr. H. Endy M. Astiwaru, MA, AAAIJ, FIIS (BPH DSN-MUI)

Adapun yang termasuk dalam Dewan Komisaris adalah sebagai berikut:

1. H. Muslim M. Yatim, Lc (Komisaris Utama)
2. Erwin Chandra Kelana, ST (Komisaris)

Adapun yang termasuk dalam Dewan Direksi adalah sebagai berikut:

1. H. Agung Yulianto, SE, Ak. M.Kom (Direktur Utama)
2. H. Rofik Hananto, SE (Direktur)
3. Supriyono, ST (Direktur)

2.5.2 Motto, Visi, Misi HPAI

HPAI memiliki Motto yaitu sebagai referensi utama produk halal dunia. Hal ini didukung oleh visi HPAI yaitu menjadi referensi utama produk halal berkualitas. Misi dari HPAI yaitu menjadi perusahaan jaringan pemasaran papan atas kebanggaan Ummat, menjadi wadah perjuangan penyediaan produk halal

10. HPAI hanya memproduksi, menyediakan, memasarkan dan mendistribusikan produk-produk yang berkualitas berdasarkan azas alamiah, ilmiah dan ilahiah.
11. Semangat keislaman menjadi spirit dan nilai-nilai akhlak serta azas dalam bekerja dan berniaga, di Halal *Network* HPAI.
12. HPAI memiliki sejumlah *Bussiness Center* (BC) dan Agen stok dalam jalur pendistribusian produk yang tersebar di hampir seluruh provinsi Indonesia, bahkan juga di beberapa negara, dan jumlah agen HPAI yang terus tumbuh berkembang.
13. HPAI menggunakan basis *online (web base) services*. Aplikasi penjualan di agen stok dan *Business Center* (BC) sudah 100 % *online*.

2.6 *Object-Oriented Analysis and Design (OOAD)*

Pada tahun 1991, Rumbaugh memperkenalkan *Object Modelling Technique* (OMT). Pendekatan yang digunakan tidak jauh berbeda dengan pendekatan yang digunakan oleh Coad Yourdon, namun dengan notasi yang berbeda. OMT tidak hanya sepenuhnya berbasis pada data driven, tetapi juga memisahkan proses dari mata dengan penggunaan *data flow diagram* yang terpisah dengan diagram kelas. OMT juga menggunakan notasi *state transition diagram* untuk memodelkan aspek dinamis sistem (Sugiarti, 2013).

Pada tahun 1994, Ivar Jacobson memperkenalkan konsep *use case* dan *object oriented software engineering* (OOSE). Pada tahun itu juga, yaitu bulan Oktober 1994 Booch, Rumbaugh dan Jacobson mempelopori usaha untuk penyatuan notasi pendekatan berorientasi objek. Pada tahun 1995, dihasilkan draft pertama, yaitu UML (versi 0.8). Sejak tahun 1996 pengembangan tersebut dikoordinasikan oleh *Object Management Group* (OMG). Pada tahun 1997, muncul UML versi 1.1 dan saat ini versi terbaru adalah UML versi 1.5 yang dirilis bulan Maret 2003. Booch, Rumbaugh dan Jacobson menyusun tiga buku serial tentang UML pada tahun 1999. Sejak saat itulah UML telah menjelma menjadi standar bahasa pemodelan untuk aplikasi berorientasi objek (Sugiarti, 2013).

2.6.1 Konsep Dasar *Object Oriented Analysis and Design (OOAD)*

Konsep OOAD mencakup analisis dan desain sebuah sistem dengan pendekatan objek, yaitu analisis berorientasi objek (OOA) dan desain berorientasi objek (OOD). OOA adalah analisis yang memeriksa *requirement* (syarat/keperluan) yang harus dipenuhi sebuah sistem dari sudut pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup perusahaan. Sedangkan OOD adalah metode untuk mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek-objek sistem dan subsistem (Sugiarti, 2013).

OOA mempelajari permasalahan dengan menspesifikasikannya dan mengobservasi permasalahan tersebut dengan menggunakan metode berorientasi objek (Sugiarti, 2013). Analisis sistem dimulai dengan adanya dokumen permintaan (*requirements*) yang diperoleh dari pihak yang berkepentingan (misalnya: klien, developer, pakar, dan sebagainya). Dokumen permintaan memiliki 2 fungsi, yaitu memformulasikan kebutuhan klien, dan membuat suatu daftar tugas. Hasil dari analisis berorientasi objek adalah deskripsi dari apa sistem secara fungsional yang diperlukan untuk dilakukan, dalam bentuk sebuah model konseptual. Hasil biasanya akan disajikan sebagai seperangkap menggunakan kasus, satu atau lebih UML diagram kelas, dan sejumlah diagram interaksi. Tujuan dari analisis berorientasi objek adalah untuk mengembangkan model yang menggambarkan perangkat lunak komputer karena bekerja untuk memenuhi seperangkat persyaratan yang ditentukan pelanggan.

OOD mengubah model konseptual yang dihasilkan dalam analisis berorientasi objek memperhitungkan kendala yang dipaksakan oleh arsitektur yang dipilih dan setiap *non-fungsional* seperti teknologi atau lingkungan, kendala seperti transaksi *throughput*, *response time*, *run – waktu platform*, lingkungan pengembangan, atau bahasa pemrograman (Sugiarti, 2013).

2.6.2 *Tools Yang Digunakan Dalam Perancangan Berorientasi Objek*

Adapun *tools* yang digunakan dalam perancangan berorientasi objek ini adalah di antaranya (Sugiarti, 2013):

1. *Object Oriented Analysis* (OOA) dan *Object Oriented Design* (OOD) dari Peter Coad dan Edward Yourdon (1990),
2. *Object Modelling Technique* (OMT) dan James Rumbaugh, Michael Blaha, William Premerlan, Frederick Eddy, dan William Lorenzen (1991),
3. *Object Oriented Software Engineering* (OOSE) dan Ivar Jacobson (1992),
4. Booch Method dan Grady Booch (1994),
5. Sritrop dan Steve Cook dan John Daniels (1994),
6. UML (*Unified Modeling Language*) dari James Rumbaugh. Grady Booch dan Ivar Jacobson (1997).

2.7 *Unified Modelling Language (UML)*

UML (*Unified Modelling Language*) adalah sebuah bahasa yang dituangkan dalam bentuk grafik atau gambar yang telah menjadi standar industri untuk visualisasi, spesifikasi, merancang atau membangun, dan mendokumentasikan dari sebuah sistem pengembangan *software* berbasis OO (*Object-Oriented*) (Sugiarti, 2013).

Pemodelan menggunakan UML merupakan metode pemodelan berorientasi objek dan berbasis visual. Kerenanya pemodelan menggunakan UML merupakan pemodelan objek yang berfokus pada pendefinisian struktur statis dan model sistem informasi yang dinamis daripada mendefinisikan data dan proses yang tujuannya adalah pengembangan tradisional. UML memberikan standar penulisan sebuah sistem *blue print*. UML juga menawarkan diagram yang dikelompokkan menjadi lima perspektif berbeda untuk memodelkan suatu sistem.

Berikut penjelasan mengenai berbagai diagram UML serta tujuannya.

2.7.1 *Use Case Diagram*

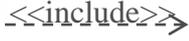
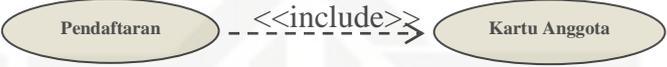
Use case diagram merupakan pemodelan untuk menggambarkan kelakuan (*behavior*) sistem yang akan dibuat (Sugiarti, 2013). Diagram *use case* mendiskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Hal yang perlu diingat, *use case diagram* bukan menggambarkan tampilan antarmuka (*user interface*), arsitektur dari sistem, kebutuhan non-fungsional, dan tujuan performansi. Sedangkan untuk penamaan *use case* adalah

nama didefinisikan sesimple mungkin, dapat dipahami, dan menggunakan kata kerja. Berikut simbol – simbol yang ada pada *use case diagram* terlihat pada tabel 2.1.

Tabel 2.1 Simbol-simbol Pada *Use Case Diagram*

| Simbol | Deskripsi |
|---|---|
|  | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit aktor; biasanya dinyatakan dengan menggunakan data kerja yang awal frase nama <i>use case</i> . |
|  | Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat, jadi walaupun simbol dari aktor itu adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor. |
|  | Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> , atau <i>use case</i> berinteraksi dengan aktor. |
|  | Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan nama <i>use case</i> yang ditambahkan, arah panah menunjuk pada <i>use case</i> yang dituju. Contoh:  |

Tabel 2.1 Simbol-simbol Pada *Use Case Diagram* (Lanjutan)

| | |
|--|--|
| <p>Include</p>  | <p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini. Ada dua sudut pandang yang cukup besar mengenai <i>include use case</i>, include berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan. Contoh:</p>  |
|--|--|

(Sumber: Sugiarti, 2013)

2.7.2 Activity Diagram

Activity diagram atau diagram aktivitas secara grafis digunakan untuk menggambarkan aliran rangkaian aktivitas baik proses bisnis maupun *use case* (Sugiarti, 2013). *Activity diagram* dapat juga digunakan untuk memodelkan *action* (aksi) yang akan dilakukan saat sebuah operasi dieksekusi, dan memodelkan hasil dari *action* tersebut.

Berikut simbol – simbol *activity diagram* terlihat pada tabel 2.2.

Tabel 2.2 Simbol-simbol Pada *Activity Diagram*

| Simbol | Deskripsi |
|---|---|
| <p><i>Activity</i></p>  | <p>Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.</p> |
| <p><i>Initial Node</i></p>  | <p>Bagaimana objek dibentuk atau diawali.</p> |
| <p><i>Activity Final Node</i></p>  | <p>Bagaimana objek dibentuk dan dihancurkan.</p> |
| <p><i>Fork Node</i></p>  | <p>Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.</p> |
| <p>Arah Pesan</p>  | <p>Menyatakan jalannya arah pesan suatu <i>activity</i>.</p> |

Tabel 2.2 Simbol-simbol Pada *Activity Diagram* (Lanjutan)

| | |
|--|---|
| <p><i>Decision</i></p>  | Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya atau tidak. |
|--|---|

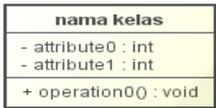
Sumber: Sugiarti (2013)

2.7.3 *Class Diagram*

Class diagram atau diagram kelas secara grafis menggambarkan struktur objek sistem. Diagram ini menunjukkan kelas objek yang menyusun sistem dan juga hubungan antara kelas objek tersebut (Sugiarti, 2013).

Berikut simbol – simbol *class diagram* terlihat pada tabel 2.3.

Tabel 2.3 Simbol-simbol Pada *Class Diagram*

| Simbol | Deskripsi |
|--|---|
| <p>Operasi</p>  | Kelas pada struktur sistem. |
| <p>Asosiasi</p>  | Relasi antara kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> . |
| <p>Asosiasi Berarah/ <i>Direction asosiation</i></p>  | Relasi antara kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> . |
| <p><i>Defedency/</i> Kebergantungan</p>  | Relasi antara kelas dengan makna kebergantungan antar kelas. |
| <p>Agregasi</p>  | Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>). |
| <p>Generalisasi</p>  | Relasi antar kelas dengan makna generalisasi-spesifikasi (umum khusus). |

(Sumber: Sugiarti, 2013)

2.8 MySQL

MySQL (*My Structure Query Language*) adalah sebuah perangkat lunak sistem manajemen basis data SQL (*Database Management System*) atau DBMS dari sekian banyak DBMS, seperti Oracle, Postagre SQL, dan lain-lain. MySQL

merupakan DBMS yang *multithread*, *multiuser* yang bersifat gratis dibawah lisensi GNU *General Public Licence* (GPL). Kelebihan MySQL yaitu dapat berjalan dengan stabil pada berbagai sistem operasi seperti windows, Linux, FreeBSD, Mac Os X Server, dan lain-lain. tidak hanya itu, MySQL memiliki kecepatan yang baik dalam menangani query (Anhar, 2010).

2.9 *Hypertext Preprocessor* (PHP)

PHP adalah bahasa pemrograman web *server side* yang bersifat *open source*. PHP merupakan script yang terintegrasi dengan HTML dan berada pada *server (server side HTML embedded scripting)*. PHP adalah *script* yang digunakan untuk membuat halaman website yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru. Semua *script* PHP dieksekusi pada *server* dimana *script* tersebut dijalankan (Anhar, 2010).

2.10 *Python*

Python adalah bahasa pemrograman yang bersifat *open source*. Bahasa pemrograman ini dioptimalisasikan untuk *software quality*, *developer productivity*, *program portability*, dan *component integration* (Lutz, 2010). *Python* telah digunakan untuk mengembangkan berbagai macam perangkat lunak, seperti *internet scripting*, *systems programming*, *user interfaces*, *product customization*, *numeric programming* dll. *Python* saat ini telah menduduki posisi 4 atau 5 bahasa pemrograman paling sering digunakan di seluruh dunia (Lutz, 2010). Bahasa pemrograman *Python* memiliki beberapa fitur yang dapat digunakan oleh pengembang perangkat lunak. Berikut adalah beberapa fitur yang ada pada bahasa pemrograman *Python* (Lutz, 2010): *Multi Paradigm Design*, *Open Source*, *Simplicity*, *Library Support*, *Portability*, *Extendable*, *Scalability*.

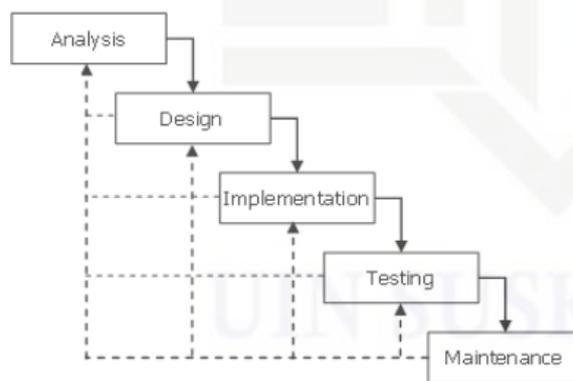
2.11 *Blackbox Testing*

Sering kali perangkat lunak mengandung kesalahan (*error*) pada perangkat lunak sering dengan *bug*. Untuk menghindari banyak *bug* maka diperlukan adanya

pengujian perangkat lunak sebelum perangkat lunak diberikan kepada *user*. *Blackbox testing* adalah pengujian perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. *Blackbox testing* dengan membuat kasus uji bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan (Graham, 1999).

2.12 Metode Pengembangan Air Terjun (*Waterfall*)

Model *Waterfall* SDLC adalah proses pengembangan perangkat lunak berurutan di mana kemajuan dianggap mengalir semakin ke bawah (mirip dengan air terjun) melalui daftar fase yang harus dijalankan agar berhasil membangun perangkat lunak komputer. Awalnya, model *Waterfall* diusulkan oleh Winston W. Royce pada tahun 1970 untuk menggambarkan praktek rekayasa perangkat lunak yang mungkin. Model *Waterfall* mendefinisikan beberapa fase berturut-turut yang harus diselesaikan satu demi satu dan pindah ke fase berikutnya hanya ketika fase sebelumnya benar-benar selesai. Untuk alasan ini, model *Waterfall* bersifat rekursif karena setiap fase dapat diulang tanpa henti sampai disempurnakan. Adapun model pengembangan *waterfall* seperti gambar berikut:



Gambar 2.4 Model Pengembangan *Waterfall*

(Sumber : Bassil, 2012)

Pada dasarnya, model *Waterfall* terdiri dari lima fase (Bassil, 2012):

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.

1. Fase Analisis

Sering dikenal sebagai Spesifikasi Kebutuhan Perangkat Lunak adalah deskripsi lengkap dan komprehensif dari perilaku perangkat lunak yang akan dikembangkan. Ini melibatkan analisis sistem dan bisnis untuk mendefinisikan baik persyaratan fungsional maupun non-fungsional.

2. Fase Desain

Ini adalah proses perencanaan dan penyelesaian masalah untuk solusi perangkat lunak. Ini melibatkan pengembang perangkat lunak dan desainer untuk menentukan rencana untuk solusi yang meliputi desain algoritma, desain arsitektur perangkat lunak, skema konseptual database dan desain diagram logis, desain konsep, desain antarmuka pengguna grafis, dan definisi struktur data.

3. Fase Implementasi

Ini mengacu pada realisasi kebutuhan bisnis dan spesifikasi desain ke dalam program yang dapat dieksekusi konkret, database, situs web, atau komponen perangkat lunak melalui pemrograman dan penyebaran. Fase ini adalah tempat kode asli ditulis dan dikompilasi menjadi aplikasi operasional, dan di mana file database dan teks dibuat. Dengan kata lain, ini adalah proses mengubah seluruh persyaratan dan cetak biru menjadi lingkungan produksi.

4. Fase Pengujian

Ini juga dikenal sebagai verifikasi dan validasi yang merupakan proses untuk memeriksa perangkat lunak solusi memenuhi persyaratan dan spesifikasi asli dan bahwa itu mencapai tujuan yang dimaksudkan. Faktanya, verifikasi adalah proses evaluasi perangkat lunak untuk menentukan apakah produk pengembangan yang diberikan fase memenuhi kondisi yang dikenakan pada awal itu tahap; sementara, validasi adalah proses evaluasi perangkat lunak selama atau pada akhir proses pengembangan untuk tentukan apakah memenuhi persyaratan yang ditentukan. Selain itu, fase pengujian adalah outlet untuk melakukan

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.

debugging di mana bug dan gangguan sistem ditemukan, dikoreksi, dan disempurnakan sesuai.

5. Fase Pemeliharaan

Ini adalah proses memodifikasi solusi perangkat lunak setelah pengiriman dan penerapan untuk menyaring output, memperbaiki kesalahan, dan meningkatkan kinerja dan kualitas. Kegiatan pemeliharaan tambahan dapat dilakukan dalam fase ini termasuk mengadaptasi perangkat lunak ke lingkungannya, mengakomodasi kebutuhan pengguna baru, dan meningkatkan keandalan perangkat lunak

