

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Berikut adalah beberapa jurnal penelitian terdahulu terkait judul penelitian skripsi ini adalah sebagai berikut :

Berdasarkan penelitian Aditya Rakhmatullah (2016) dengan judul “ Penerapan *Knowledge Management System* Di Dinas Pertanian Cianjur Menggunakan *Cf-Idf* Dan *Vector Space Model*” Apabila data tanaman itu tidak terdokumentasi dengan baik para pengguna yang mencari informasi mengenai data tanaman merasa kesulitan. Maka itu terbentuklah *Knowledge Management System* (KMS) pengolahan data tanaman menggunakan algoritma CF-IDF dan VSM. Sistem ini menggunakan konsep KMS (*Knowledge Management System*) untuk mengelola pengetahuan berupa data tanaman, dan juga menggunakan algoritma CF-IDF (*Concept Frequency - Inverse Document Frequency*) dan VSM (*Vector Space Model*). Maka menggunakan algoritma tersebut agar memudahkan pengguna dalam mencari informasi berkaitan dengan kata kunci yang dimasukkan pada mesin pencarian.

Berdasarkan penelitian yang dilakukan oleh Cahyo Adi Suprpto, Setiawan Assegaff (2017) dengan judul “Analisis Dan Perancangan *Knowledge Management System* Pada Sma Negeri 6 Kota Jambi” Tujuan penelitian ini adalah untuk merancang sebuah sistem yang mendukung adanya pendokumentasian dan penyebaran *Knowledge* yang lebih baik keseluruhan

pendidik dan tenaga kependidikan di SMAN 6 Kota Jambi. Bagaimana merancang suatu *Knowledge Management System* yang dapat mentransfer *Knowledge* dari dokumen, panduan dan solusi dari kasus yang pernah terjadi agar dapat didistribusikan ke seluruh Guru dan Tenaga Kependidikan secara merata. Faktor yang akan diteliti yaitu komponen utama dari *Knowledge Management* (KM) yaitu *people*, *process*, dan *technology* dan untuk perancangan *Knowledge Management System* menggunakan teori *Knowledge roadmap* dari Tiwana.

Berdasarkan penelitian Husni Faqih (2015) dengan judul “Pengembangan *Knowledge Management System* pada Perusahaan *Multifinance*: Studi Kasus Pada *Bess Finance* Cabang Slawi” Hasil Penelitian *Knowledge Management* tidak hanya menyangkut penyediaan infrastruktur *Knowledge Management System* yang memfasilitasi pertukaran *knowledge* antar personil organisasi, tetapi harus diimbangi dengan membangun budaya yang melekat dalam kehidupan organisasi. Budaya tersebut adalah budaya *knowledge learning and sharing*. *Knowledge management* merupakan sebuah strategi yang penting dalam upaya meningkatkan nilai organisasi. Peningkatan nilai organisasi yang diharapkan setelah penerapan *Knowledge Management System* di *BESS Finance* Cabang Slawi adalah meningkatnya kualitas SDM dan arus informasi antar personil menjadi lebih baik karena hasil dari budaya *knowledge sharing* yang hasilnya adalah membuat peningkatan kepercayaan konsumen terhadap perusahaan.

II.2. Landasan Teori

II.2.1. Sistem

Definisi sistem adalah “kumpulan dari bagian-bagian yang bekerja sama untuk mencapai tujuan yang sama.” Definisi sistem adalah “sekumpulan objek-objek yang saling berelasi dan berinteraksi serta hubungan antar objek bisa dilihat sebagai suatu kesatuan yang dirancang untuk mencapai suatu tujuan. Sistem adalah penggabungan dari bagian-bagian atau komponen-komponen yang terpisah-pisah dan disatukan menjadi satu rangkaian dan menjadi suatu fungsi yang baru (Aris : 2015).

Kata sistem berasal dari bahasa Latin (*systema*) dan bahasa Yunani (*sustema*) yang artinya adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi. Secara umum sistem adalah kumpulan dari beberapa bagian tertentu yang saling berhubungan secara harmonis untuk mencapai suatu tujuan tertentu. Elemen-elemen yang mewakili suatu sistem secara umum adalah masukan (*Input*) pengolahan (*Processing*) dan keluaran (*Output*). (Priyo Sutopo,dkk 2016).

II.2.2. Informasi

Informasi dapat diibaratkan sebagai darah yang mengalir didalam tubuh manusia, seperti halnya informasi di dalam sebuah perusahaan yang sangat penting untuk mendukung kelangsungan perkembangannya, Sehingga terdapat alasan bahwa informasi sangat dibutuhkan bagi sebuah perusahaan. Akibat bila kurang mendapatkan informasi, dalam waktu tertentu perusahaan akan mengalami

ketidakmampuan mengontrol sumber daya, sehingga dalam mengambil keputusan-keputusan strategis sangat terganggu, yang pada akhirnya akan mengalami kekalahan dalam bersaing dengan lingkungan pesaingnya. Disamping itu, sistem informasi yang dimiliki seringkali tidak dapat bekerja dengan baik. Masalah utamanya adalah bahwa sistem informasi tersebut terlalu banyak informasi yang tidak bermanfaat atau berarti sistem terlalu banyak data. (Andri Kristanto : 2018 : 7)

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

1. Akurat (*Accurate*)

Berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya.

2. Tepat waktu (*Timelines*)

Berarti informasi yang datang pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai logika karena informasi merupakan landasan dalam pengambilan keputusan.

3. Relevan (*Relevance*)

Berarti informasi tersebut bermanfaat bagi pemakainya (Deppi Linda, 2016 : 62-63).

II.4. Sistem Informasi

Sistem Informasi adalah sekumpulan prosedur yang terorganisir dan dijalankan guna dapat menyediakan informasi untuk menunjang atau mendukung organisasi. Sistem informasi memiliki definisi yang berbeda menurut para ahli, namun secara umum, sistem informasi adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen. Istilah ini digunakan untuk merujuk tidak hanya pada penggunaan organisasi teknologi informasi dan komunikasi (TIK), tetapi juga untuk cara di mana orang berinteraksi dengan teknologi ini dalam mendukung proses bisnis. (Nuryana : 2016 : 2)

II.5. Knowledge

Knowledge Management adalah usaha untuk meningkatkan pengetahuan yang berguna dalam organisasi, diantaranya membiasakan budaya berkomunikasi antar personil, memberikan kesempatan untuk belajar, dan menggalakan saling berbagi *knowledge*. Dimana usaha ini akan menciptakan dan mempertahankan peningkatan nilai dari inti kompetensi bisnis dengan memanfaatkan teknologi informasi yang ada. Hal ini disarikan dari pendapat McInerney sebagai berikut:

Knowledge management system merupakan strategi untuk meningkatkan efektifitas dan peluang/kesempatan pengembangan kompetensi. Beberapa hal yang perlu dilakukan untuk menumbuhkan budaya berbagi pengetahuan diantaranya:

- a) Menciptakan *know-how* dimana setiap pegawai berkesempatan dan bebas menentukan cara baru untuk menyelesaikan tugas dan berinovasi serta peluang untuk mensinergikan pengetahuan eksternal kedalam institusi.
- b) Menangkap dan mengidentifikasi pengetahuan yang dianggap bernilai dan direpresentasikan dengan cara yang logis.
- c) Penempatan pengetahuan yang baru dalam format yang mudah diakses oleh seluruh pegawai dan pejabat.
- d) Pengelolaan pengetahuan untuk menjamin kekinian informasi agar dapat direview untuk relevansi dan akurasi.
- e) Format pengetahuan yang disediakan di portal adalah format yang user friendly agar semua pegawai dapat mengakses dan mengembangkan setiap saat. (Lita Wulantika: 2016)

II.2. Knowledge Management System

Knowledge Management System (KMS) merupakan *System* yang dibangun untuk mengelola manajemen pengetahuan (*Knowledge Management*) karena manajemen pengetahuan merupakan suatu hal yang sangat luas dan sangat kompleks. KMS adalah *System* dari KM (*Knowledge Management*). Menurut *Knowledge Management System* (KMS) merupakan teknologi yang memungkinkan KM untuk berjalan dengan efektif dan efisien. Definisi *Knowledge Management System* (KMS) yaitu sekelompok *System* informasi yang diaplikasikan untuk mengelola *knowledge* yang terdapat di dalam organisasi. *Knowledge Management System* (KMS) merupakan *System* yang berbasiskan

teknologi informasi yang dikembangkan untuk mendukung proses-proses inti dari *Knowledge Management System* (KMS) yaitu, penciptaan *knowledge* (*knowledge creation*), penyimpanan *knowledge* (*knowledge storage*), pemindahan *knowledge* (*knowledge transfer*), dan pengaplikasian *knowledge* tersebut (*knowledge application*) dalam organisasi. Adanya penekanan terhadap tujuan utama dari *Knowledge Management System* (KMS) untuk meningkatkan keefektifan organisasional dengan adanya manajemen sistem terhadap sebuah pengetahuan. (Anderias Eko Wijaya : 2017)

II.2.6. Normalisasi

Normalisasi adalah proses pembentukan struktur basis data sehingga sebagian besar ambiguity bisa dihilangkan. Normalisasi merupakan sebuah teknik dalam logical desain sebuah basis data relasional yang mengelompokkan atribut dari suatu tabel sehingga membentuk struktur tabel yang normal. Adapun kriteria tabel dikatakan normal adalah ketika tidak ada kerangkapan data (redundansi data). Tujuan dari normalisasi adalah untuk :

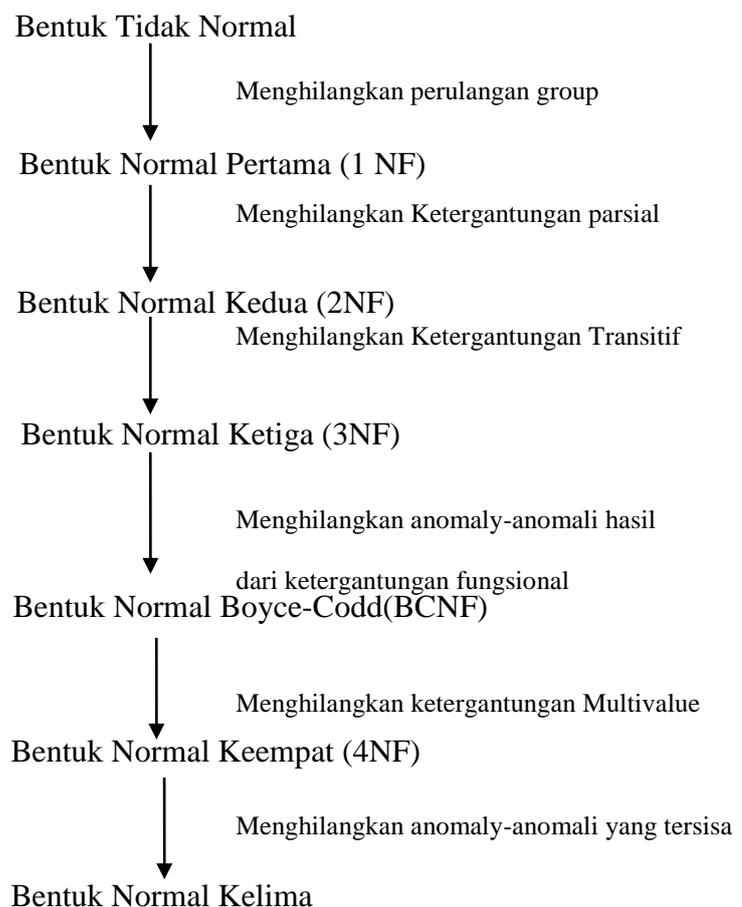
1. Untuk menghilangkan kerangkapan data sehingga meminimumkan pemakaian *storage* yang dipakai oleh *base relations* (file)
2. Untuk mengurangi kompleksitas
3. Untuk mempermudah pemodifikasian data

Gambaran proses normalisasi adalah :

1. Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu ke beberapa tingkat, kemudian

2. Apabila tabel yang diuji belum memenuhi persyaratan tertentu, maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal.

Untuk melakukan proses tersebut dibutuhkan beberapa tahapan. Tahapan dalam normalisasi dimulai dari tahap paling ringan (1NF) hingga paling ketat (5NF). Biasanya hanya sampai pada tingkat 3NF atau BCNF karena sudah cukup memadai untuk menghasilkan tabel-tabel yang berkualitas baik. Urutan tahapan normalisasi tampak seperti gambar II.1.



Gambar. II.1 Tahapan Pada Normalisasi
(Sumber : Dwi Puspitasari, dkk ; 2016)

Adapun aturan dalam normalisasi adalah suatu tabel dikatakan baik (efisien) atau normal jika memenuhi 3 kriteria sbb:

1. Jika ada dekomposisi (penguraian) tabel, maka dekomposisinya harus dijamin aman (*Lossless-Join Decomposition*). Artinya, setelah tabel tersebut diuraikan / didekomposisi menjadi tabel-tabel baru, tabel-tabel baru tersebut bisa menghasilkan tabel semula dengan sama persis.
2. Terpeliharanya ketergantungan fungsional pada saat perubahan data (*Dependency Preservation*).
3. Tidak melanggar Boyce-Codd Normal Form (BCNF)

Jika kriteria ketiga (BCNF) tidak dapat terpenuhi, maka paling tidak tabel tersebut tidak melanggar Bentuk Normal tahap ketiga (3rd Normal Form / 3NF). Pada penelitian ini formula yang dibuat sampai memenuhi bentuk normal ke 3 yaitu 3NF. (Dwi Puspitasari, 2016)

II.2.7. XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Nama *XAMPP* merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam *GNU General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis.

Untuk mendapatkannya dapat men-*download* langsung dari *web* resminya. (Randi V Palit, 2015 : 2).

II.2.8. PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah web dan bisa digunakan pada dokumen HTML.PHP dirancang untuk dapat bekerja sama dengan database server dan dibuat sedemikian rupa sehingga pembuatan dokumen HTML yang dapat mengakses database menjadi begitu mudah. Tujuan dari bahasa scripting ini adalah untuk membuat aplikasi dimana aplikasi tersebut yang dibangun oleh PHP pada umumnya akan memberikan hasil pada web browser, tetapi proses secara keseluruhan dijalankan di server. (Saipul Anwar, 2016)

II.2.9. MySql

MySQL (*My Structure Query Language*) merupakan sebuah program pembuat database yang bersifat *open source*, artinya semua orang dapat menggunakannya dan dapat dijalankan pada semua *platform* baik *windows* maupun *linux*. *MySQL* juga merupakan sebuah perangkat lunak sistem manajemen basis data *SQL* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi multi *user*. *MySQL* juga sering dikenal dengan nama sistem manajemen *database* relasional. Suatu *database* relasional menyimpan data dalam table yang terpisah. Tabel -table tersebut terhubung oleh suatu relasi terdefinisi yang memungkinkan

Imemperoleh kombinasi data dari beberapa table dalam suatu permintaan. Untuk administrasi *database*, seperti pembuatan *database*, pembuatan tabel, dan sebagainya dapat digunakan aplikasi berbasis web seperti *phpMyAdmin* dengan aplikasi *XAMPP*. (Saipul Anwar, 2016)

II.2.10. UML (*Unified Modeling Language*)

Unified Modelling Language (UML) merupakan salah satu bentuk *language* atau bahasa, menurut pencetusnya UML didefinisikan sebagai bahasa *visuai* untuk menjelaskan, memberikan spesifikasi, merancang, membuat model, dan mendokumentasikan aspek aspek dari sebuah sistem. Perancangan sistem pada UML adalah sebagai berikut:

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasiskan UML adalah sebagai berikut :

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 dari dengan sistem informasi yang akan dibuat. 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.

Tabel II.2. Tabel Simbol *Use Case* Diagram

Gambar	Keterangan
	Use Case, Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

	Actor, Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data..
	<i>Include</i> , Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
	<i>Extend</i> , Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.

(Sumber : Ade Hendini; 2016: 109-110)

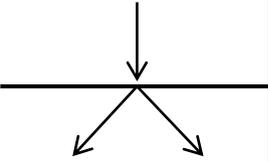
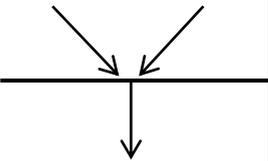
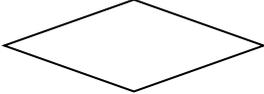
2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram (Diagram Aktifitas) menggambarkan berbagai alir aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Diagram Aktifitas merupakan *state* diagram khusus, di mana sebagian besar state adalah action dan sebagian besar transisi di-*trigger* oleh selesainya state sebelumnya (*internal processing*). Oleh karena itu Diagram AKtifitas tidak menggambarkan *behaviour* internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Menggambarakan proses bisnis dan urutan aktifitas dalam sebuah proses.

Dipakai pada *business*

Tabel II.3. Tebel Simbol *Activity Diagram*

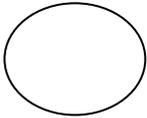
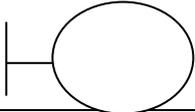
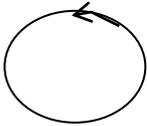
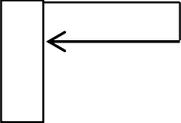
Gambar	Keterangan
	<i>Start point</i> , Bagaimana objek dibentuk atau diawali.
	<i>End point</i> , Bagaimana objek dibentuk dan dihancurkan
	<i>Activites</i> , Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
	<i>Fork</i> (Percabangan), Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran
	<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

(Sumber : Ade Hendini; 2016: 109-110)

3. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram*, yaitu. :

Tabel II.4. Tabel Simbol *Sequence Diagram*

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>formentry</i> dan <i>form</i> cetak
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber : Ade Hendini; 2016: 3)

4. *Class Diagram* (Diagram Kelas)

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class*

diagram secara khas meliputi: Kelas (*Class*), Relasi, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/Method*), *Visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau *cardinality*.

Tabel II.5. Tabel *Multiplicity Class Diagram*

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Ade Hendini; 2016: 111)