

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Teori Dasar**

##### **2.1.1 Pengertian Sistem**

*Systema* merupakan bahasa latin sistem yang diartikan sebagai terintegrasinya kumpulan elemen (Rusmawan, 2019, p. 28). Sistem dapat diartikan sebagai banyaknya komponen yang tersusun dan terhubung secara teratur (Baskoro & Gamariyah, 2022, p. 23). Sekelompok elemen yang saling berinteraksi adalah sistem. Sistem terdiri dari komponen, periode sistem, *interface*, *input* dan *output* (Nursanty et al., 2022).

Elemen yang terdapat dalam sistem yaitu masukan, pengolahan, keluaran dan umpan balik (*feedback*) yang ditujukan sebagai mekanisme pengendalian (Rusmawan, 2019, p. 31). Sistem dapat ditinjau dari beberapa segi, seperti sistem abstrak, fisik, alamiah, buatan manusia, sistem tertentu, tidak tertentu, tertutup dan terbuka. Sistem tertentu (*deterministic system*) merupakan sistem yang berjalan berdasarkan prediksi. Sistem berisikan kumpulan objek berupa sumber daya, orang dan prosedur untuk memenuhi perencanaan yang sudah dirancang (Ginting et al., 2022, pp. 2–3).

##### **2.1.2 Pengertian Informasi**

Sekumpulan data yang dapat dikelola sebagai bahan untuk mengambil keputusan disebut dengan informasi (Alimin et al., 2022, p. 35). Informasi berisikan fakta, data dan lainnya. Informasi bertujuan untuk memperluas pemahaman dalam suatu organisasi (Nursanty et al., 2022, p. 11). Dalam suatu organisasi peran informasi digunakan agar organisasi tersebut tetap berjalan dan bersinergi (Ginting et al., 2022).

Informasi memiliki siklus didalamnya yaitu *input*, proses dan *output*. *Input* pada siklus informasi berisikan kumpulan data yang saling terkait sebagai bahan untuk proses. Siklus kedua informasi adalah proses yang digunakan untuk merincikan operasi aritmatika terhadap data yang dimasukkan dan melakukan proses lainnya yang dibutuhkan untuk menghasilkan informasi. Siklus informasi ketiga yaitu *output* berupa informasi yang bermanfaat bagi pengguna (Pratama et al., 2020). Kualitas dari informasi dapat dibuktikan apabila mencakup karakteristik informasi seperti dapat dengan mudah diakses oleh semua orang, informasi harus memenuhi kebutuhan semua orang, *relevance* dan *secure* (Nugroho, 2017, p. 37).

### 2.1.3 Pengertian Sistem Informasi

Pengolahan data dengan menggunakan mesin berupa komputer disebut dengan sistem informasi. Sumber utama dari sistem informasi diantaranya perangkat keras (*hardware*) dapat berupa komputer maupun sejenisnya, perangkat lunak (*software*) untuk melakukan pengolahan data agar menjadi suatu informasi dan manusia (*brainware*) yang bertugas untuk mengoperasikan sumber lainnya agar menghasilkan informasi yang akurat. Data yang berisikan fakta juga dibutuhkan untuk kemudian dikelola. Jaringan digunakan untuk menyebarkan informasi yang dihasilkan dari pengolahan data (Ardiansah & Ahmad, 2021, pp. 8–9).

Manfaat dari sistem informasi bagi perusahaan yaitu dapat meningkatkan efisiensi operasional, pekerja dapat menghemat waktu dengan adanya sistem informasi. Perusahaan dengan mudah mengenalkan inovasi dalam bisnisnya melalui sistem informasi. Sumber informasi yang lebih signifikan dapat membangun perusahaan melalui sistem informasi. Pengambilan keputusan dapat dibantu dengan adanya sistem informasi (Nugroho, 2017).

#### 2.1.4 *Reminder* Kegiatan

*Reminder* kegiatan bertujuan untuk mengingatkan kembali para peserta. Tim pelaksana akan memberikan *reminder* pada satu hari sebelum dimulainya kegiatan. Informasi yang terdapat pada *reminder* kegiatan yang diberikan oleh tim pelaksana berisikan nama kegiatan, waktu kegiatan, tempat kegiatan, detail tempat dan lain sebagainya yang dapat membantu para peserta (Zaini & Aransyah, 2021, p. 131). *Daily reminder* dapat suatu individu untuk mengorganisir kegiatan agar terjadwal dan meningkatkan rasa produktivitas dalam dirinya (Widyastuti, 2021, p. 202). *Time management* dapat diperbaiki dengan menerapkan salah satu strategi yaitu menggunakan aplikasi *reminder*. Penggunaan *reminder* membantu mengingatkan seseorang terhadap tugas dan kegiatan yang dimilikinya (Indra, 2019, p. 241).

#### 2.1.5 Dokumentasi Kegiatan

Dokumentasi ditafsirkan sebagai sarana informasi yang datanya tercatat dapat digunakan untuk belajar, sebagai bukti, penelitian dan sejenisnya. Proses pencatatan, penyimpanan informasi berupa fakta yang memiliki makna dalam suatu kegiatan merupakan bagian dari dokumentasi. Informasi tertulis maupun informasi yang dihasilkan melalui elektronik dapat dijadikan sebagai dokumentasi (Rahmi, 2022).

Pekerjaan yang berkaitan dengan pengumpulan dan penyebaran informasi merupakan bagian dari dokumentasi (Rodin, 2021, p. 17). Kegiatan yang terjadi akan dijadikan sebagai dokumentasi dalam bentuk foto, video dan bentuk dokumentasinya lainnya (Japar et al., 2018, p. 25). Tujuan dari dokumentasi yaitu sebagai alat komunikasi dan bahan yang dapat dipertanggungjawabkan (Tasalim et al., 2022, p. 14). Dokumentasi dapat digunakan sebagai gambaran secara nyata mengenai kegiatan yang sudah dilakukan (Fitria et al., 2021, p. 129).

### 2.1.6 Tata Kelola

Suatu Perguruan Tinggi perlu membuat tata kelola yang baik guna meningkatkan kinerjanya. Tata kelola yang dibantu oleh sistem wajib menyentuh semua area yang terdapat pada Perguruan Tinggi tersebut (Rahayu, 2019, p. 124). Perusahaan yang baik tentunya memiliki tata kelola yang dapat mencegah maupun meringankan risiko yang terjadi. Tata kelola merupakan suatu rangkaian yang berisikan proses, etiket, peraturan dan strategi yang terdapat pada suatu lembaga untuk mengarahkan, mengelola dan mengontrol (Sudarmanto et al., 2021, p. 123).

### 2.1.7 *System Development Life Cycle (SDLC)*

Pengembangan sistem dilakukan secara terorganisir untuk mencapai hasil yang memenuhi kebutuhan pengguna. Pengembangan sistem dapat disebut dengan *System Development Life Cycle* (Ikatan Bankir Indonesia, 2017). SDLC merupakan proses menafsirkan sistem informasi yang mampu menunjang bisnis melalui perancangan sistem. Empat fase SDLC adalah perencanaan, analisis, desain dan implementasi. Terdapat tahapan dari masing-masing fase. Berikut penjabaran dari keempat fase SDLC (Dennis et al., 2020).

#### a. Perencanaan (*planning*)

Tahapan pertama dalam pembuatan dan pengembangan suatu sistem. Pada tahap perencanaan digunakan untuk memahami tujuan dari dibangunnya sistem informasi pada suatu organisasi. Step yang perlu dilakukan dalam tahap perencanaan yaitu menganalisis kelayakan dan mengontrol arahnya proyek melalui fase SDLC. Analisis kelayakan dalam tahap perencanaan berisikan kelayakan teknis ide, kelayakan ekonomi dan kelayakan organisasi (Dennis et al., 2020, p. 4).

b. Analisis (*analysis*)

Tahap analisis bertujuan untuk mengevaluasi sistem yang sedang dipakai. Hasil analisis dapat dijadikan sebagai bahan dalam mengambil keputusan dan menyusun pengembangan dari sistem. Dalam tahap analisis terdapat beberapa kegiatan untuk mendukung tujuan dari tahapan ini. Kegiatan dalam tahapan analisis yaitu meneliti sistem berjalan, mengumpulkan kebutuhan sistem baru dan merancang sistem baru (Santi, 2020, p. 26).

Analisis sistem berjalan dilaksanakan untuk mengetahui perkara yang terjadi pada sistem saat ini dan acuan untuk merancang sistem baru. Pengumpulan kebutuhan sistem baru yang dilakukan melalui wawancara atau kuesioner dengan tujuan sebagai gambaran operasi bisnis pada sistem baru yang akan dikembangkan. Perancangan sistem baru berisikan keputusan persetujuan pengguna sistem dan persyaratan bisnis yang harus dipenuhi pada sistem baru yang dicantumkan menjadi proposal sistem (Dennis et al., 2020, p. 5).

c. Desain (*design*)

Tahap desain merupakan perancangan sistem yang berhubungan dengan spesifikasi rancangan. Hasil akhir dari tahapan desain yaitu rancangan keluaran, masukan, *platform*, basis data dan rancangan kontrol (Asrory & Safitriani, 2021, p. 653). Langkah pertama yang dilakukan dalam tahapan desain yaitu memverifikasi mengenai sistem yang akan dikembangkan. Langkah kedua yaitu melakukan pengembangan desain arsitektur dasar dengan menggambarkan perangkat dan infrastruktur yang digunakan. Penentuan lokasi basis data dan spesifikasi dari sistem yang akan dikembangkan. Langkah terakhir yaitu melakukan pengembangan pada sistem dengan mendefinisikan program yang digunakan pada tahap implementasi (Dennis et al., 2020, p. 5).

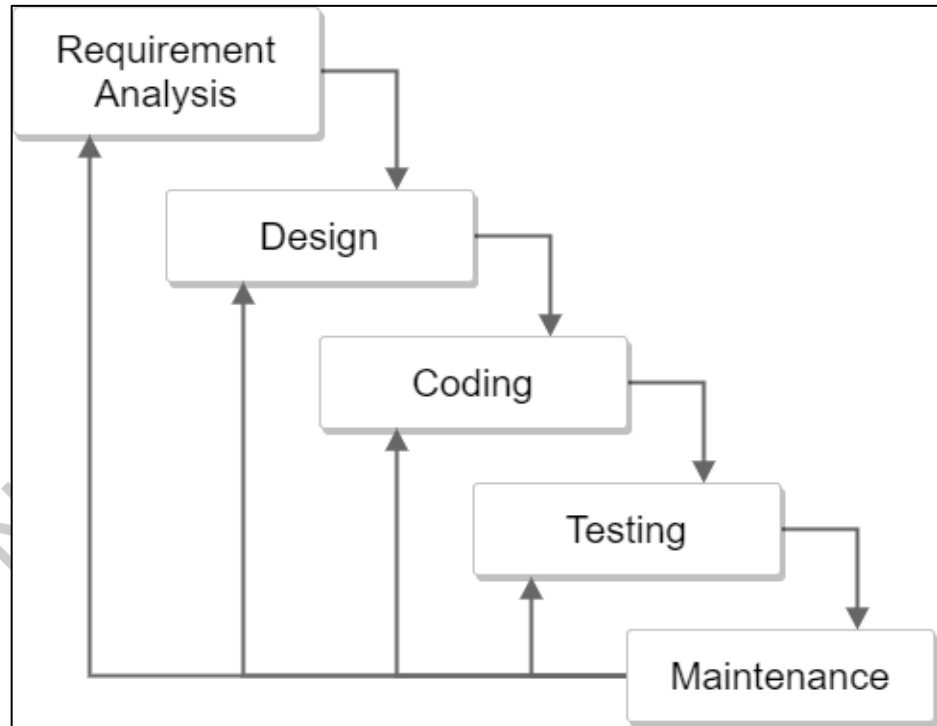
d. Implementasi (*implementation*)

Pada tahap implementasi baru dimulainya proses pengkodean sistem informasi dengan menggunakan bahasa program agar dapat dijalankan oleh mesin (Asrory & Safitriani, 2021, p. 654). Terdapat tiga langkah dalam tahap implementasi yaitu pertama membangun sistem melalui bahasa pemrograman dan pengujian sistem yang dilakukan bersamaan guna menghindari bug yang kemungkinan besar. Langkah kedua yaitu konversi dan training pengguna, pada saat melakukan konversi dapat memilih antara tiga strategi berikut, pertama *cutover* langsung, konversi paralel dan konversi bertahap. Langkah ketiga yaitu penetapan rencana dukungan untuk sistem yang mencakup tinjauan pasca implementasi dan memberikan buku pedoman pemakaian sistem kepada pengguna (Dennis et al., 2020, p. 6).

### 2.1.8 Metode *Waterfall*

Pengembangan sistem memerlukan adanya metodologi untuk membantu dalam pengerjaannya. Penentuan memilih metodologi yang tepat tentunya tidak mudah. Metodologi dalam pengembangan sistem tidak ada satupun yang menjadi pilihan terbaik. Setiap metodologi tentunya memiliki kelebihan dan kekurangan (Dennis et al., 2020, p. 18).

Metode *waterfall* dapat diterapkan dalam pengembangan dengan kebutuhan pengguna yang jelas, menerapkan teknologi yang familiar dengan pengguna, dapat diandalkan, jangka waktu yang tidak pendek dan sudah direncanakan (Dennis et al., 2020, p. 18). Metode *waterfall* digambarkan seperti aliran air terjun. Terdapat lima tahapan pada metode *waterfall* yaitu perencanaan, analisis, desain, implementasi dan *maintenance* (Rasch, 2019). Pada **Gambar 2.1** dapat dilihat lima tahapan *waterfall*.



Gambar 2.1 Metode Waterfall (Saxena, 2020)

### 2.1.9 Object Oriented Analysis and Design (OOAD)

OOAD adalah prinsip rekayasa dengan melalui analisis dan desain untuk membuat produk, sistem dan lain sebagainya. OOAD dalam penerapannya memakai fasilitas pemodelan berupa *Unified Model Language* (UML). UML merupakan pemaparan dari struktur dan perilaku sistem yang dideskripsikan dalam bentuk visual berupa diagram. Diagram pada UML memiliki tiga kategori. Setiap kategori didalamnya terdapat beberapa diagram sebagai berikut (Yang, 2021).

- a. *Structure diagram* dapat berupa *class diagram*, *object diagram*, *component diagram* dan *package diagram*.
- b. *Behavior diagram* berupa *use case diagram*, *activity diagram* dan *state machine diagram*.
- c. *Interaction diagram* seperti *sequence diagram*, *communication diagram* dan *interaction overview diagram*.

Terdapat aturan yang perlu diketahui dalam pembuatan sistem menggunakan OOAD. Diarahkan oleh *usecase* dan arsitektur akan memusatkan. OOAD bersifat perulangan dengan perbaikan sedikit demi sedikit.


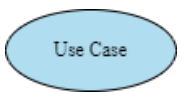

a. *Use Case Diagram*

*Use case* menjabarkan interaksi yang dikerjakan aktor dengan sistem. Penamaan harus dipaparkan mudah dipahami. Dalam pembuatan *use case* perlu diperhatikan dua hal utama berupa aktor dan *use case* (Sukanto & Shalahuddin, 2018).

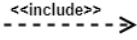
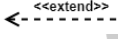
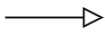
- 1) Aktor merupakan pengguna, proses atau sistem lain yang ikut berinteraksi.
- 2) *Use case* menampilkan fungsional dalam sistem.

Interaksi yang dijalankan oleh aktor dengan sistem ialah berupa kegiatan. Kegiatan yang digambarkan melalui *use case* dipaparkan dengan menggunakan simbol. Simbol dalam pembuatan *use case* diagram ditampilkan pada **Table 2.1** (Azis et al., 2019).

**Table 2.1 Simbol Use Case Diagram**

| Simbol   | Nama Simbol        | Keterangan   |
|--|--------------------|--|
| <br>Actor, Role | <i>Actor</i>       | Pemakai, <i>hardware</i> dan sejenisnya yang memiliki akses. |
| <br>Use Case    | <i>Use Case</i>    | Kegiatan yang dijalankan oleh <i>actor</i> .                 |
|                 | <i>Association</i> | Interaksi <i>actor</i> dengan <i>use case</i> .              |







| Simbol  | Nama Simbol           | Keterangan   |
|---|-----------------------|--|
|  | <i>Include</i>        | Ikatan antara <i>use case</i> yang dimiliki oleh <i>use case</i> lainnya.                      |
|  | <i>Extends</i>        | Aktivitas yang ditunjukkan oleh panah <i>extends</i> tidak wajib dilakukan oleh <i>actor</i> . |
|  | <i>Generalization</i> | Hubungan generalisasi antar <i>actor</i> .   |

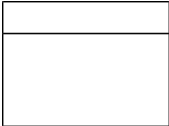

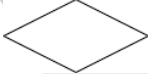
Sumber: (Azis et al., 2019, p. 379)

#### b. Activity Diagram

Diagram yang memvisualkan aksi dari proses pada sistem. Dalam pembuatan *activity* diagram membutuhkan *symbol* yang masing-masing memiliki fungsi. Simbol pada *activity* diagram digunakan untuk memudahkan pembacanya guna mengetahui kerja dari sistem. Simbol pada *activity* diagram ditampilkan pada **Table 2.2** (Harianto et al., 2019).

**Table 2.2 Simbol Activity Diagram**

| Simbol  | Nama Simbol                | Keterangan  |
|---|----------------------------|---|
|  | <i>Activity</i>            | Menampilkan tindakan yang berkaitan.  |
|  | <i>Initial Node</i>        | Penanda awal dimulainya aktivitas pada sistem   |
|  | <i>Activity Final Node</i> | Akhir dari kegiatan pada sistem.  |
|  | <i>Fork Node</i>           | Aliran yang dapat digunakan untuk menandakan bahwa terdapat kegiatan yang digabungkan menjadi satu. |

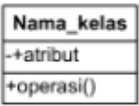

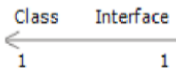
| Simbol  | Nama Simbol             | Keterangan   |
|---|-------------------------|--|
|  | <i>Swimlane</i>         | Pemisah antar penanggung jawab terhadap kegiatan yang dilakukan pada sistem. |
|  | <i>State Transition</i> | Aliran dari aktivitas dan menggabungkan beberapa kegiatan.                   |
|  | <i>Decision</i>         | Cabang yang menunjukkan bahwa terdapat lebih dari satu kondisi.              |




Sumber: (Harianto et al., 2019, pp. 19–20)

### c. Class Diagram

Dalam membangun sistem dibutuhkan adanya pembuatan gambaran dari struktur sistem yang didefinisikan berupa bentuk kelas. Penerapan *class* diagram dapat membantu untuk mendeskripsikan masing-masing objek dalam sistem yang saling berhubungan. *Class* diagram memiliki simbol dalam pembuatannya. Pada **Table 2.3** menampilkan simbol dari *class* diagram (Azis et al., 2019).

Table 2. 3 Simbol Class Diagram

| Simbol  | Nama Simbol        | Kesimpulan  |
|---|--------------------|---|
|  | <i>Class</i>       | Sekumpulan objek berisikan atribut dan operasi.           |
|  | <i>Package</i>     | <i>Folder</i> yang digunakan untuk menampung objek-objek. |
|  | <i>Association</i> | Objek yang memiliki relasi dengan objek lain.             |

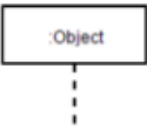

| Simbol  | Nama Simbol        | Kesimpulan  |
|---|--------------------|---|
|  | <i>Dependency</i>  | Objek yang bergantung dari objek lain. Apabila terdapat method atau operasi pada suatu objek dijalankan pada objek lainnya. |
|  | <i>Inheritance</i> | Objek yang merupakan turunan dari suatu objek lainnya untuk mengakses   |
|  | <i>Visibility</i>  | Informasi akses yang berupa <i>public</i> , <i>private</i> , <i>protected</i> dan lain sebagainya.                          |

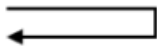




Sumber: (Azis et al., 2019, pp. 383–384)

d. *Sequence* Digram

*Sequence* diagram menggambarkan perilaku dari *object* yang terdapat pada *use case*. Deskripsi waktu pengiriman dan penerimaan pesan dapat digambarkan melalui *sequence* diagram. Pembuatan *sequence* diagram agar sesuai dengan ketentuan diperlukannya pengetahuan mengenai simbol pada *sequence* diagram. Simbol dalam pembuatan *sequence* diagram ditampilkan pada **Table 2.4** (Harianto et al., 2019).

**Table 2.4** Simbol *Sequence* diagram

| Simbol  | Nama Simbol           | Kesimpulan  |
|---|-----------------------|---|
|  | <i>Lifelane</i>       | objek <i>entry interface</i> saling berkaitan           |
|  | <i>Object Message</i> | Alur kontak antar objek berupa informasi dari aktivitas |

| Simbol  | Nama Simbol              | Kesimpulan   |
|---|--------------------------|--|
|    | <i>Message (Return)</i>  | Menyatakan bahwa alur feedback atau kembali dalam satu objek <i>lifelane</i> |
|    | <i>Message (Return)</i>  | Menyatakan bahwa alur kembali antar <i>lifelane</i>                          |
|    | <i>Activation</i>        | Menyatakan keadaan aktif pada objek  |
|   | <i>Message (Destroy)</i> | Objek yang mematikan objek lainnya melalui panah diarahkan ke objek terakhir |
|  | <i>Actor</i>             | Peran pengguna ketika berinteraksi dengan sistem                             |

Sumber: (Harianto et al., 2019, pp. 20–21)

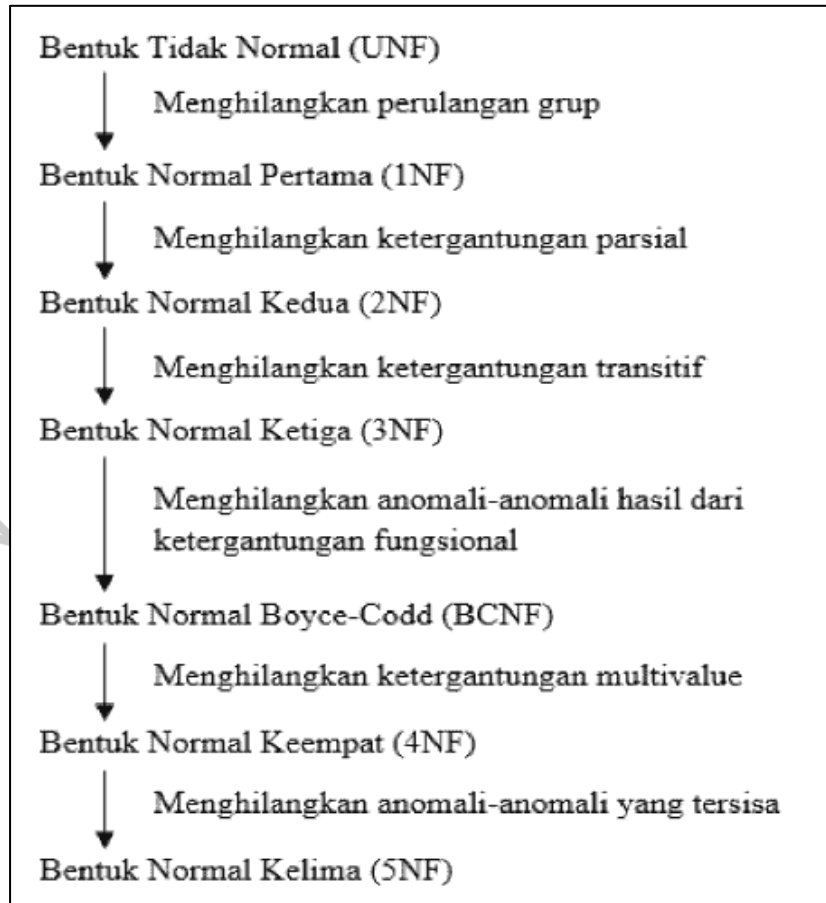
### 2.1.10 Perancangan Basis Data

Bagian krusial dari sistem informasi merupakan basis data. Basis data digunakan sebagai rujukan untuk membuat dan mengembangkan suatu program pada zaman modern. Perancangan basis data harus dibuat sedemikian rupa agar program dapat berjalan lebih cepat. Basis data memiliki manfaat bagi penggunanya dalam hal kecepatan proses, dapat dipakai bersamaan, *maintenance* data, menghemat *space*, keakuratan data, kelengkapan informasi dan keamanan informasi. Basis data berperan penting dalam pengolahan data karena basis data merupakan dasar penyedia informasi dan penentu kualitas informasi yang akurat (Rusmawan, 2019).

*Database* selalu disangkutpautkan dengan basis data. Sistem digunakan untuk mengatur dan menyimpan data agar lebih mudah dan aman (Kadarsih & Pujianto, 2022, p. 2). Basis data merupakan media penyimpanan yang dapat diakses lebih mudah. *Database Management System* (DBMS) adalah teknik yang diterapkan guna melakukan penyimpanan, pengaturan dan mengemukakan data. DBMS memiliki peran yang dapat memudahkan penggunaannya yaitu mengelola data, memelihara data, mengelola akses data, pemindahan data, mencegah konflik yang terjadi dalam pembaharuan data, mengelola transaksi pada data, mendukung bahasa *query*, mengawasi pada saat *backup* data dan menjaga keamanan data (Sukanto & Shalahuddin, 2018).

Sistem basis data pada DBMS memerlukan komponen-komponen pendukung seperti, *hardware*, *software*, data, prosedur dan manusia. Keberhasilan suatu sistem basis data tergantung pada struktur yang dirancang dan manusia. Basis data dapat dinyatakan memiliki struktur yang baik apabila sudah memenuhi dua kriteria. Kriteria pertama yaitu terdapat redundansi yang seminim mungkin (Hadiprakoso, 2021, pp. 14–18).

Ditanyakan baik yang kedua yaitu datanya memiliki *record* dalam relasi yang dapat disisipkan, dimodifikasi dan dihapus. Struktur yang baik juga dapat diverifikasi melalui normalisasi data. Normalisasi adalah tahapan yang mengelompokkan komponen data menjadi suatu *table* yang masing-masing memiliki entitas (Suendri, 2021, p. 94). Tujuan dari normalisasi yaitu meminimalisir redundansi data, memecahkan *table* yang ada menjadi beberapa *table* dan mempercepat proses *Create, Read, Update and Delete* (CRUD). Normalisasi data memiliki 5 sampai dengan 6 tahapan. Pada umumnya normalisasi yang sudah mencapai 3NF atau BCNF dapat dinyatakan sudah cukup baik dan sudah cukup memadai (MF, 2020). Pada **Gambar 2.2** dapat dilihat kegiatan yang dilakukan dalam setiap tahapan normalisasi dari 1NF sampai dengan 5NF.



Gambar 2.2 Tahapan Normalisasi (Anggoro et al., 2021)

Bentuk normalisasi pertama (1NF) dapat dinyatakan jika tidak adanya atribut yang berulang dan pada *table* memiliki atribut *key* sebagai relasi antar *table*. Normalisasi kedua (2NF) yaitu memisahkan ketergantungan fungsional yang terdapat pada *table* normalisasi pertama dan setiap *table* hanya diperbolehkan bergantung pada satu kunci. Normalisasi ketiga (3NF) adalah membentuk sebuah *table* yang mana kolom dari *table* diambil dari *primary key* yang telah dipisahkan sesuai kaidah (Suendri, 2021, pp. 96–98). Normalisasi *Boyce-Codd* (BCNF) hampir sama dengan 3NF yaitu menghilangkan ketergantungan data yang berulang. Normalisasi keempat (4NF) membuat setiap *multivalued dependencies* (MVD) harus merupakan *functional dependencies*. Normalisasi kelima (5NF) apabila tidak ada *lossless decomposition* pada tabel-tabel yang lebih kecil, dibentuknya 5NF berdasarkan *join dependency* (Anggoro et al., 2021).

Data yang sudah dilakukan normalisasi data selanjutnya perlu dibangun untuk melakukan pengelolaan *database*. Bahasa yang standar digunakan untuk mengakses *database* yaitu *Structure Query Language (SQL)*. SQL dapat digunakan untuk berinteraksi dan mengendalikan suatu *database*. *Syntax* yang terdapat dalam *SQL* mudah dimengerti karena berbasis Bahasa Inggris (Perdana, 2020).

Pengembangan *SQL* awalnya berdasarkan teori kalkulus dan aljabar relasional. *SQL* memiliki tiga jenis perintah. *Data Definition Language (DDL)* adalah Perintah untuk mengidentifikasi struktur *database* seperti, membuat, mengubah dan mengganti nama *table*. *Data Manipulation Language (DML)* yang berkaitan terhadap pengelolaan data pada tabel seperti memilih, menyisipkan, memperbaharui dan menghapus data. *Data Control Language (DCL)* berkaitan dengan pengelolaan pengguna beserta hak akses seperti, *grant* dan *revoke* (Solichin, 2016, p. 99).

Perintah *SQL* membutuhkan *server* untuk mendukung kinerjanya salah satunya melalui *MySQL*. Penggunaan *MySQL* dapat melakukan perintah *SQL* yang sangat variatif. *MySQL* dapat digunakan terhadap aplikasi kecil maupun besar. *MySQL* memiliki kelebihan tersendiri didalamnya seperti, berjalan konstan pada berbagai sistem operasi, keamanannya berlapis, *multi-user* dan dapat mendeteksi pesan kesalahan (Rusmawan, 2019).

Pengaksesan *MySQL* memerlukan *web server* berupa *XAMPP*. *XAMPP* merupakan *software* sumber terbuka yang dikembangkan oleh tim *Apache*. *Software XAMPP* cocok dalam lingkungan pengujian proyek dengan menggunakan *MySQL*, *PHP*, *Apache* dan *Perl*. Instalasi *XAMPP* sudah termasuk *database MySQL* dengan menggunakan aplikasi *phpMyAdmin*. *PhpMyAdmin* termasuk *software* yang penerapannya menggunakan web dengan tujuan mengatur *database* pada *MySQL* dan *MariaDB* melalui *interface* grafis (Yudhanto & Prasetyo, 2019).

### 2.1.11 Interaksi Manusia Dan Komputer

Interaksi berisikan *input*, proses dan *output*. Siklus interaksi dilakukan dengan menetapkan tujuan dan kegiatan yang harus dilakukan oleh kedua objek melalui implementasi antarmuka yang sudah dibuat. Manusia adalah komponen utama dalam interaksi manusia dan komputer yang bertugas sebagai pengguna. Kemampuan dan kualifikasi manusia menghasilkan desain produk yang dapat memenuhi kebutuhan manusia. Komputer bertujuan sebagai alat untuk memasukkan perintah atau keinginan yang dilakukan oleh manusia melalui *keyword* (Dalle et al., 2019).

Pengembangan interaksi manusia dan komputer memiliki beberapa faktor seperti, faktor RPL, AI, linguistik komputasional, psikologi, multimedia, antropologi dan matematika. Faktor Rekayasa Perangkat Lunak (RPL) berisikan tahapan awal dari sebuah aspek produksi perangkat lunak seperti, analisa, desain, *code*, pengujian dan pemeliharaan sistem dengan tujuan untuk membuat program menjadi lebih efektif dan mudah digunakan. Faktor *Artificial intelligence* (AI) adalah kecerdasan buatan yang terdapat pada sistem dengan pola pikir seperti manusia. Faktor linguistik komputasional digunakan untuk memudahkan dalam penggunaannya. Faktor psikologi bertujuan untuk membantu dalam merancang program yang *user friendly*. Faktor multimedia untuk menghasilkan tampilan yang menarik. Faktor antropologi berisikan cara kerja dari masing-masing kelompok. Faktor matematika memudahkan proses perhitungan yang terdapat dalam sistem (Haryati & Feranika, 2021).

## 2.2 Tinjauan Studi

Pendukung *outline* rencana penelitian skripsi memerlukan adanya tinjauan studi. Tinjauan studi berkaitan dengan topik yang dibahas. Berikut beberapa referensi tinjauan studi.



1. **RANCANG BANGUN APLIKASI PENJADWALAN KEGIATAN AKADEMIK DISERTAI SISTEM *REMINDER* BERBASIS *RESPONSIVE WEB DESIGN***, penelitian dilakukan oleh (Ramsari & Rifaldi, 2018). Jurnal FIKI: Jurnal Teknologi Informasi dan Komunikasi, Volume IX No. 1, Mei 2018. Jurnal ini membahas perihal masalah yang terdapat pada Fakultas Ilmu Komputer dan Informatika di Universitas Nurtanio Bandung mengenai pencatatan dan pengingat kegiatan akademik yang masih menggunakan cara-cara konvensional. Masalah yang terdapat pada jurnal ini yaitu tidak adanya sistem pencatatan dan pengingat kegiatan sehingga menyebabkan pejabat fakultas dan dosen sering tidak mengikuti rapat bahkan sampai terlambat memberikan hasil evaluasi belajar mahasiswa. Sehingga dirancanglah penjadwalan aktivitas perguruan yang digunakan dalam membantu proses pencatatan dan pengingat aktivitas akademik. Sistem pencatatan kegiatan akademik diharapkan dapat mengoptimalkan efektifitas aktivitas akademik pada Fakultas Ilmu Komputer dan Informatika di Universitas Nurtanio Bandung. Dengan adanya sistem yang dirancang melalui analisis sistem dan *usecase* yang baik, tentunya dapat mempermudah dosen dan pejabat fakultas dalam mengetahui kegiatan yang akan diselenggarakan.

2. **RANCANG BANGUN SISTEM INFORMASI NOTULEN RAPAT (STUDI KASUS: FAKULTAS ILMU KOMPUTER UNIVERSITAS LANCANG KUNING)**, penelitian yang dilakukan oleh (Van FC et al., 2020). ZONasi: Jurnal Sistem Informasi, Vol 2 No. 2, September 2020. Jurnal ini membahas permasalahan yang terjadi pada Fakultas Ilmu Komputer di Universitas Lancang Kuning yaitu masih menggunakan cara manual dalam melakukan pencatatan dan membagikan notulensi hasil rapat yang berupa notulensi. Notulensi hasil rapat pada Fakultas Ilmu Komputer di Universitas Lancang Kuning masih menggunakan buku untuk mencatat yang memungkinkan catatan atau laporan tersebut mudah hilang tentunya sangat besar. Fakultas

Ilmu Komputer di Universitas Lancang Kuning juga tidak memiliki sistem pendukung untuk membagikan notulensi hasil rapat. Sehingga dibuatlah rancang bangun sistem informasi notulensi rapat yang dapat membantu dosen dan karyawan pada Fakultas Ilmu Komputer di Universitas Lancang kuning. Rancangan dibangun sebagai pengingat dari isi agenda rapat sebelumnya. Penyimpanan laporan atau catatan berupa notulensi hasil rapat dirasa aman karena akan disimpan pada *database* yang berupa *MySQL*. Rancangan dari *use case* dan tampilan sistem yang terdapat pada masing-masing tampilan dari halaman tentunya dapat membantu para dosen dan karyawan dari Fakultas Ilmu Komputer di Universitas Lancang Kuning dalam menggunakan aplikasi tersebut.

- 3. PENGEMBANGAN SISTEM MANAJEMEN NOTULENSI DAN DOKUMENTASI RAPAT BERBASIS WEB (STUDI KASUS: JURUSAN TEKNIK INFORMATIKA FAKULTAS ILMU KOMPUTER UNIVERSITAS BRAWIJAYA)**, penelitian yang dilakukan oleh (Nugraha et al., 2020). Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, Volume 4 No. 4 hal. 1273-1280. Jurnal ini membahas mengenai tata kelola rapat pada jurusan Teknik Informatika di Universitas Brawijaya. Peserta yang berstatus tidak dapat hadir maka tidak akan menerima hasil sesi, dokumentasi tidak diarsipkan sehingga menimbulkan kebingungan bagi peserta rapat tersebut dan manajemen kegiatan sesi yang kurang baik. Pengembangan sistem manajemen berupa dokumentasi rapat dengan notulensi dan dokumentasi diperlukan untuk mempermudah ketua pelaksana rapat, sekretaris rapat, sekretaris Unit Mutu Jaminan (UMJ) dan peserta rapat seperti dosen, karyawan maupun anggota organisasi yang ada dalam mengetahui hasil rapat. Adapun mempermudah dalam melakukan akreditasi program studi. Sistem memiliki fungsi utama yaitu untuk membuat agenda, membuat notula, dan tempat penyimpanan dokumentasi berupa video rapat. Sistem yang

dikembangkan sudah mendapatkan skor mencapai 76,6% dalam pengujian *usability* dengan menggunakan metode *System Usability Scale* (SUS), skor tersebut menandakan maupun menyatakan bahwa sistem yang dikembangkan sudah mencukupi kebutuhan dari penggunanya.

**4. SISTEM INFORMASI MANAJEMEN RAPAT BERBASIS WEB**

**PADA SMK SAHID JAKARTA**, penelitian yang dilakukan oleh (Wisnu Wijaya & Nurcahya, 2022). Jurnal Jurnal VISUALIKA, Vol.8 No.1, April 2022, hal. 26 – 39. Jurnal ini membahas SMK Sahid Jakarta yang tidak memiliki pencatatan rencana agenda kegiatan rapat secara lebih jelas. Informasi kegiatan rapat yang sudah dilaksanakan hanya dicatat melalui buku notulen dan dokumentasi kegiatan rapat masih disimpan secara manual. Masalah tersebut menyebabkan proses administrasi dan penyimpanan berkas kegiatan masih belum tersusun dengan rapih dan tersentral. SMK Sahid Jakarta membutuhkan adanya sistem informasi pertemuan berupa daring untuk mengorganisasikan undangan kegiatan dan mendokumentasikan hasil kegiatan. Sistem informasi manajemen rapat dengan berbasis web di SMK Sahid Jakarta diharapkan dapat membantu dan memudahkan pengguna sistem yaitu *admin*, panitia dan peserta rapat dalam mengelola kegiatan rapat dalam sekolah. Dengan fitur unggulan yang diberikan seperti pendeteksi ruangan rapat dan fitur integrasi hasil rapat. Adapun dengan penggunaan *database* yang berupa *MySQL* membantu menyimpan data lebih aman dan dapat memudahkan pengguna maupun operator dalam mencari data surat melalui jaringan internet dimanapun dan kapanpun data tersebut dibutuhkan.

**5. IMPLEMENTASI E-ARSIP PADA PROGRAM STUDI TEKNIK**

**INFORMATIKA**, penelitian dilakukan oleh (Dedi Irawan & Simargolang, 2018). Jurnal Teknologi Informasi, Vol.2 No.1, Juni 2018. Jurnal ini membahas masalah pada Program Studi Teknik

Informatika di Universitas Asahan mengenai pengelolaan dokumen yang masih menggunakan kertas, akibatnya membuat tugas menjadi tidak efektif dan tidak efisien. Maka dibutuhkan adanya sistem *e-arsip* yang digunakan untuk membantu kaprodi dan sekretaris dalam melakukan pengarsipan dokumen. Oleh karena itu, proses dari pencarian dan arsip data menjadi lebih akurat. Dengan mengimplementasi aplikasi *e-arsip* pada Program Studi Teknik Informatika di Universitas Asahan diharapkan dapat dengan mudah menginput surat hanya dengan *scan*, meminimalkan terjadinya rusaknya dokumen penting, dan pencarian berkas menjadi lebih mudah.

