

## 2. LANDASAN TEORI

### 2.1. Sistem Informasi

Eni Endaryati (2021) mendefinisikan sistem informasi sebagai “suatu sistem yang berfungsi untuk mengorganisasi formulir, catatan dan laporan yang dikoordinasi untuk menghasilkan informasi yang dibutuhkan dalam pembuatan keputusan manajemen dan pimpinan perusahaan.”

Menurut Krismiaji (2015), sistem informasi adalah “suatu sistem yang memproses data dan transaksi untuk menghasilkan informasi yang bermanfaat untuk merencanakan, mengendalikan, dan mengoperasikan bisnis.”

Sistem informasi memanfaatkan teknologi komputer dalam mengelola data atau transaksi perusahaan menjadi suatu informasi yang tepat, akurat, dan relevan dalam pengambilan keputusan.

### 2.2. Pengertian Gaji

Sugiyarso dan Winarni (2005) berpendapat bahwa gaji merupakan “sejumlah pembayaran yang diberikan kepada pegawai yang melakukan tugas administratif dan manajemen yang biasanya ditetapkan per bulan.” Sedangkan upah merupakan “imbalan yang biasanya diberikan kepada buruh yang melaksanakan pekerjaan kasar mengutamakan kekuatan fisik, jumlah upah yang dibayarkan biasanya ditetapkan secara harian atau berdasarkan unit pekerjaan yang diselesaikannya.” (Sugiyarso & Winarni, 2005)

Menurut Sujarweni (2015), gaji adalah “pembayaran atas jasa-jasa yang telah dilaksanakan oleh karyawan yang dilakukan oleh perusahaan setiap bulan.”

### 2.3. Pengertian Presensi

Menurut KBBI (Kamus Besar Bahasa Indonesia) presensi diartikan sebagai kehadiran. (*Presensi*, 2023).

Fungsi presensi dalam perusahaan adalah mencatat kehadiran karyawan, dasar perhitungan gaji karyawan, melihat sisa cuti karyawan, dan membuat pencatatan administrasi menjadi lebih rapi.

Jenis-jenis sistem presensi yang digunakan oleh perusahaan yaitu (*Presensi: Pengertian, Jenis Dan Kelebihannya Untuk Perusahaan*, 2022):

a. Sistem Presensi *Fingerprint*

Sistem ini mengidentifikasi dan memverifikasi karyawan berdasarkan data sidik jarinya. Sistem ini merupakan salah satu sistem presensi yang banyak digunakan oleh perusahaan. Sistem ini dapat meminimalisir terjadinya kecurangan seperti titip absen.

b. Sistem Presensi *Magnetic Card*

Sistem ini menggunakan kartu magnetik yang ditempelkan ke mesin *scanner* untuk mencatat karyawan berdasarkan identitas yang tercantum pada kartunya. Selain digunakan sebagai tanda kehadiran, kartu ini juga dapat digunakan sebagai tanda pengenal. Kelemahan dari sistem ini adalah kartu identitas dapat dititipkan ke orang lain.

c. Sistem Presensi *Password*

Sistem ini menggunakan kata sandi yang berbeda-beda yang dimiliki oleh setiap karyawan untuk mencatat kehadiran. Kelemahan dari sistem ini sama seperti sistem kartu yaitu kata sandi dapat dititipkan ke karyawan lain. Selain itu, karyawan tidak dapat mengidentifikasi kehadiran jika ia melupakan kata sandinya.

d. Sistem Presensi *Online*

Sistem presensi *online* memungkinkan karyawan untuk mencatat kehadirannya dengan menggunakan teknologi internet. Sistem ini akan menyimpan data kehadiran karyawan di dalam *database* berbasis *cloud*. Sistem ini cocok untuk karyawan yang bekerja dari rumah atau WFO (*work from home*).

#### 2.4. **Entity Relationship Diagram (ERD)**

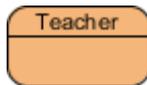
Menurut salah satu pakar Brady dan Loonam dalam artikelnya (2010) menjelaskan bahwa *Entity Relationship Diagram* (ERD) merupakan “teknik yang digunakan analisis sistem dalam tahap analisis kebutuhan dari proyek pengembangan sistem untuk mewakili kebutuhan data dari sebuah organisasi.” Meskipun pada dasarnya merupakan teknik pembuatan diagram atau alat visual, ERD memberikan dasar untuk desain *relational database* yang menjadi landasan sistem informasi yang sedang dikembangkan. ERD beserta informasi pendukungnya membentuk model data yang selanjutnya digunakan sebagai spesifikasi untuk *database*.

Menurut Setiawan (2021) dalam sebuah ERD terdapat empat komponen utama untuk memodelkan suatu sistem. Komponen-komponen tersebut akan dijelaskan sebagai berikut:

- Entitas

Entitas dalam ERD merupakan objek atau konsep yang dapat didefinisikan dalam suatu sistem. Dalam model ER, entitas ditampilkan sebagai persegi panjang dengan sudutnya

yang melengkung dan nama entitas terletak di atas, sedangkan atribut-atributnya terletak di dalam badan entitas. Sebuah entitas juga dapat menjadi entitas kuat atau entitas lemah. Entitas kuat memiliki identifikasi (*primary key*) dan tidak bergantung pada entitas lain apapun untuk keberadaannya. Entitas lemah adalah entitas yang bergantung pada entitas kuat untuk keberadaannya. Hal ini berarti entitas lemah memiliki *foreign key* ke entitas lain. Simbol entitas dapat dilihat pada Gambar 2.1. sebagai berikut:



Gambar 2.1. Entitas

Sumber: *What is Entity Relationship Diagram (ERD)?* (n.d.). Visual Paradigm.

<https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/>

- Atribut

Atribut adalah properti atau karakteristik dari entitasnya. Sebuah atribut memiliki nama yang menggambarkan properti tersebut dan tipe yang menjelaskan jenis atribut yang dimilikinya, misalnya *varchar* untuk *string*, dan *int* untuk bilangan bulat. Ada beberapa jenis atribut yang biasa digunakan dalam ERD, berikut adalah jenis-jenisnya:

- Atribut Kunci

Atribut kunci atau *Key Attributes* adalah atribut yang berfungsi untuk menentukan data yang bersifat penting. Atribut kunci bersifat unik dan biasanya berbentuk angka atau numerik. Contoh: NPWP, NIM (Nomor Induk Mahasiswa).

- Atribut Sempel

Atribut simpel atau *Simple Attributes* adalah atribut yang tidak dapat dipecah lagi dan bernilai tunggal. Contoh: alamat, tahun terbit buku, nama penerbit.

- Atribut Multinilai

Atribut multinilai atau *Multivalue Attributes* adalah atribut yang memiliki nilainya lebih dari satu. Contoh: nama beberapa pengarang dari sebuah buku pelajaran.

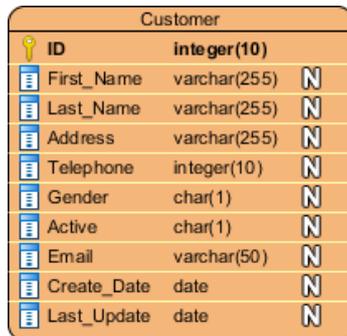
- Atribut Gabungan

Atribut gabungan atau *Composite Attributes* adalah atribut yang terdiri dari beberapa atribut yang berukuran lebih kecil dan memiliki arti tertentu. Contoh: nama lengkap yang terbagi menjadi nama depan, tengah, dan belakang.

- Atribut Derivatif

Atribut derivatif atau *Derived Attributes* adalah atribut yang dihasilkan dari atribut lain dan atributnya tidak wajib untuk ditulis dalam ERD. Contoh: usia, kelas, selisih harga.

Simbol entitas dengan beberapa atribut di dalamnya dapat dilihat pada Gambar 2.2. sebagai berikut:



Gambar 2.2. Entitas dengan Atribut

Sumber: *What is Entity Relationship Diagram (ERD)?* (n.d.). Visual Paradigm.

<https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/>

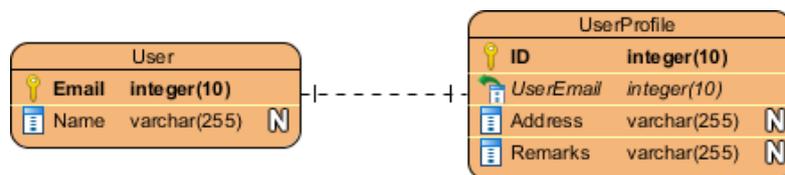
- Relasi

Hubungan antara dua entitas atau lebih menandakan bahwa entitas-entitas tersebut terkait satu sama lain dengan cara tertentu. Relasi sendiri sering disebut dengan proses.

Terdapat tiga jenis relasi yang digunakan dalam ERD yaitu:

- *One-to-one*

*One to one* berarti setiap entitas hanya dapat memiliki relasi dengan satu entitas lain. Contoh: data mahasiswa dengan NIM (Nomor Induk Siswa).



Gambar 2.3. *One-to-one relationship*

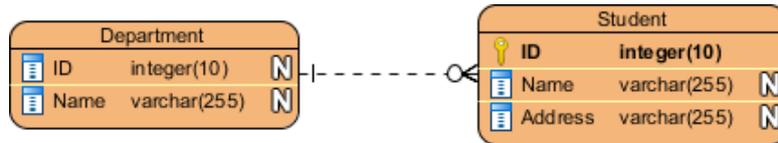
Sumber: *What is Entity Relationship Diagram (ERD)?* (n.d.). Visual Paradigm.

<https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/>

- *One-to-many*

*One to many* berarti satu entitas dapat memiliki relasi dengan beberapa entitas, begitu pula sebaliknya. Contoh: jurusan dengan mahasiswa, wali kelas dengan

murid.

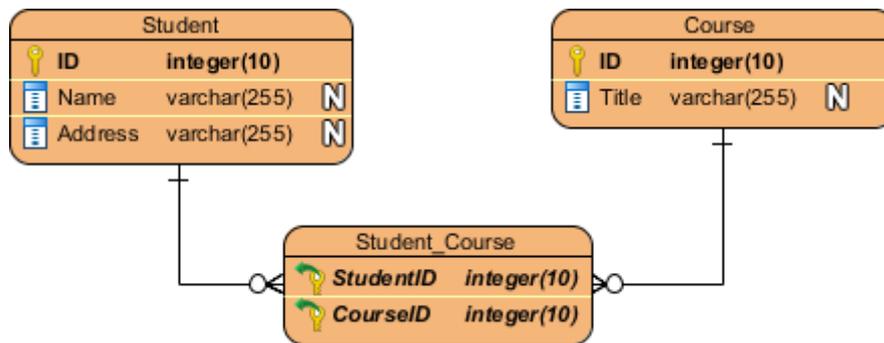


Gambar 2.4. One-to-many relationship

Sumber: *What is Entity Relationship Diagram (ERD)?* (n.d.). Visual Paradigm. <https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/>

- o *Many-to-many*

*Many to many* berarti setiap entitas yang ada dapat memiliki relasi dengan entitas lain, begitu pula sebaliknya. Contoh: mahasiswa dengan data terkait UKM (Unit Kegiatan Mahasiswa).



Gambar 2.5. *Many-to-many relationship*

Sumber: *What is Entity Relationship Diagram (ERD)?* (n.d.). Visual Paradigm. <https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/>

- Kardinalitas

Kardinalitas mendefinisikan jumlah kemungkinan kejadian pada suatu entitas yang terkait dengan jumlah kejadian pada entitas lain. Kardinalitas biasanya diekspresikan sebagai angka, tetapi juga bisa berupa simbol tergantung pada gaya diagram yang digunakan. Nilai kardinalitas yang umum adalah nol, satu, atau banyak. Tiga hubungan kardinal yang umum adalah *one-to-one*, *one-to-many*, dan *many-to-many*.

Sebuah model ER umumnya digambarkan dalam hingga tiga tingkat:

1. Model Data Konseptual

Model data konseptual memodelkan objek bisnis yang harus ada dalam suatu sistem dan hubungan antara mereka. Model konseptual dikembangkan untuk menyajikan gambaran keseluruhan dari sistem dengan mengenali objek bisnis yang terlibat.



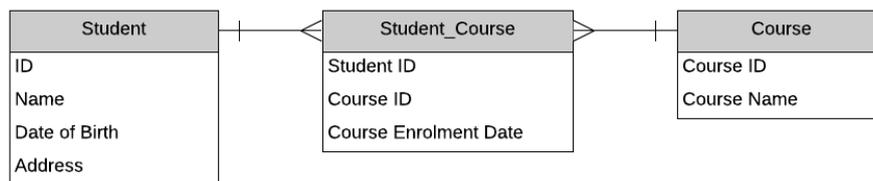
Gambar 2.6. Model Data Konseptual

Sumber: *A Guide to the Entity Relationship Diagram (ERD)*. (2022). Database Star.

<https://www.databasestar.com/entity-relationship-diagram/>

## 2. Model Data Logis

Model data logis adalah versi rinci dari model data konseptual. Model ini dikembangkan untuk memperkaya model konseptual dengan mendefinisikan secara eksplisit kolom-kolom dalam setiap entitas dan memperkenalkan entitas-operasional dan transaksional.



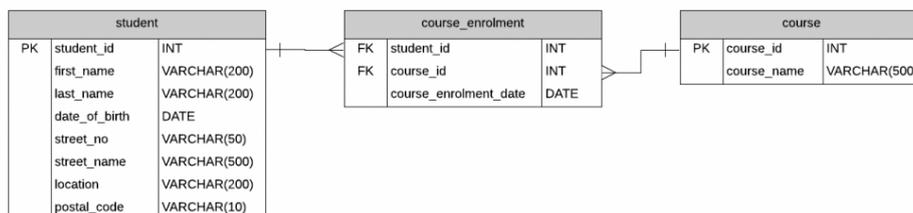
Gambar 2.7. Model Data Logis

Sumber: *A Guide to the Entity Relationship Diagram (ERD)*. (2022). Database Star.

<https://www.databasestar.com/entity-relationship-diagram/>

## 3. Model Data Fisik

Model data fisik mewakili desain *blueprint* sebenarnya dari *relational database*. Model data fisik memperjelas model data logis dengan menetapkan setiap kolom dengan tipe, panjang, boleh *null*, dll.



Gambar 2.8. Model Data Fisik

Sumber: *A Guide to the Entity Relationship Diagram (ERD)*. (2022). Database Star.

<https://www.databasestar.com/entity-relationship-diagram/>

## 2.5. Data Flow Diagram (DFD)

Menurut Rosa A. S. dan M. Shalahuddin (2018) *Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah “representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).”

Terdapat dua jenis DFD yaitu logika dan fisik. DFD logika menunjukkan bagaimana data bergerak melalui suatu sistem untuk melaksanakan fungsi bisnis tertentu. Pelaksanaan dari DFD logika ditampilkan dalam DFD fisik.

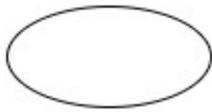
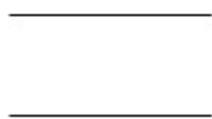
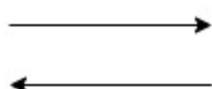
Menurut Angky (2022) terdapat empat simbol dasar yang digunakan dalam *Data Flow Diagram* yaitu:

- Proses atau *Process*  
Proses mewakili sebuah alur untuk mengubah *input* menjadi *output* dalam konten atau format berbeda. Proses bisa sederhana seperti mengumpulkan data dan menyimpannya di *database*, atau bisa juga rumit seperti menghasilkan laporan yang berisi penjualan bulanan dari semua toko retail di wilayah tertentu.
- Penyimpanan Data atau *Data Store*  
Penyimpanan data merupakan tempat menyimpan seluruh data untuk digunakan kembali pada proses selanjutnya.
- Entitas Eksternal atau *External Entity*  
Entitas eksternal merupakan pihak ketiga atau komponen di luar sistem informasi yang menyediakan data ke sistem atau menerima *output* dari sistem. Entitas eksternal mewakili bagaimana sistem informasi berinteraksi dengan dunia luar.
- Aliran Data atau *Data Flow*  
Aliran data menggambarkan perpindahan data dari satu bagian sistem informasi ke bagian lainnya. Aliran data dapat mewakili elemen data tunggal seperti ID pelanggan, atau dapat mewakili kumpulan elemen data. Terdapat beberapa aturan di dalam aliran data, yaitu:
  - Entitas tidak dapat memberikan data ke entitas lain tanpa melalui proses.
  - Data tidak dapat berpindah langsung dari entitas ke penyimpanan data tanpa melalui proses.
  - Data tidak dapat dipindahkan langsung dari penyimpanan data tanpa melalui proses.

- o Data tidak dapat berpindah langsung dari satu penyimpanan data ke penyimpanan lainnya tanpa melalui proses.

Tabel 2.1.

Notasi Data Flow Diagram

Notasi	Keterangan
	Proses/ <i>Process</i>
	Penyimpanan Data/ <i>Data Store</i>
	Entitas Eksternal/ <i>External Entity</i>
	Aliran Data/ <i>Data Flow</i>

Menurut Rosa A.S dan M. Shalahuddin (2016) langkah-langkah perancangan dengan menggunakan DFD adalah sebagai berikut:

1. Membuat DFD level 0 atau *Context Diagram*. DFD level 0 memberikan gambaran keseluruhan dan merupakan level tertinggi dalam menggambarkan sistem yang akan dibuat. Diagram konteks hanya berisi satu proses yang mewakili seluruh sistem.
2. Membuat DFD level 1. DFD level 1 menguraikan DFD level 0 yang sebelumnya sudah dibuat. DFD level 1 menggambarkan detail dari aktivitas pemrosesan yang ada dalam sistem yang akan dikembangkan.
3. Membuat DFD level 2. Modul-modul pada DFD level 1 dapat diuraikan menjadi DFD level 2. Penguraian modul lebih detail bergantung pada tingkat perincian modul tersebut. Apabila modul tersebut sudah cukup detail maka modul tersebut sudah tidak perlu diuraikan lebih lanjut. Dalam sebuah sistem, jumlah DFD level 2 konsisten dengan jumlah modul pada DFD level 1 yang diuraikan.

## 2.6. Tinjauan Studi

### 2.6.1. Penelitian Sistem Penggajian Sejenis

Penelitian sistem penggajian yang pernah dilakukan sebelumnya berjudul “Sistem Informasi Penggajian Karyawan Berbasis Java *Desktop*”. Permasalahan yang diangkat dalam penelitian ini adalah sistem penggajian yang dilakukan di beberapa tempat usaha di wilayah Jakarta Selatan dan sekitarnya secara manual dengan menggunakan kertas sebagai media pencatatan memungkinkan terjadinya kesalahan *input*, rentan terhadap manipulasi, sulit untuk mencari data, dan efektivitas pencatatan masih kurang maksimal. Penelitian ini berfokus pada perancangan sistem informasi penggajian berbasis *desktop* dengan menggunakan bahasa pemrograman Netbeans IDE 8.1 sebagai lingkungan pengembangan aplikasinya untuk meningkatkan efektivitas dan efisiensi kinerja maupun waktu. (Rahmawati & Yaumaidzinnaimah, 2021)

Penelitian sistem penggajian lainnya yang pernah dilakukan sebelumnya berjudul “Rancang Bangun Sistem Informasi Penggajian Dosen Berbasis Web”. Permasalahan yang diangkat dalam penelitian ini adalah sistem informasi penggajian di Universitas Garut tergabung ke dalam Sistem Informasi Akademik (SIMAK), namun *server* Sistem Informasi Akademik tersebut sering kali mengalami *down* yang diakibatkan banyaknya pengunjung atau pengakses sistem informasi sehingga proses penggajian pun terhambat dari kendala *server* SIMAK. Penelitian ini berfokus pada perancangan sistem informasi penggajian berbasis *website* khusus dosen dengan menggunakan bahasa pemrograman PHP dan *framework laravel* untuk mempermudah perhitungan dan perekapan laporan gaji serta menghindari tercampurnya data penggajian dosen dengan data yang lain. (Mulyani et al., 2021)

Perbedaan penelitian yang sudah dilakukan dengan penelitian yang akan dilakukan oleh penulis adalah sistem penggajian ini dibuat secara khusus dengan menyesuaikan kebutuhan di CV Trustline Indonesia yang membutuhkan sistem penggajian yang terintegrasi dan sistem presensi yang lebih baik. Selain itu, teknologi yang digunakan pada penelitian penulis saat ini berbasis *website*, berbeda dengan teknologi yang digunakan pada penelitian di wilayah Jakarta yang berbasis *desktop*. Sistem penggajian yang akan dibuat oleh penulis juga berbeda dengan sistem penggajian di Universitas Garut yang fokusnya hanya untuk dosen saja, sedangkan penulis akan membuat sistem penggajian untuk seluruh karyawan di CV Trustline Indonesia.

### 2.6.2. Penelitian Sistem Presensi Sejenis

Penelitian sistem presensi yang pernah dilakukan sebelumnya berjudul “Perancangan Sistem Informasi Absensi Guru dan *Staff* pada SMK Pancakarya Tangerang berbasis Web”. Permasalahan yang diangkat dalam penelitian ini adalah sistem presensi guru dan staf yang sedang berjalan saat ini masih dilakukan secara manual dengan pencatatan pada buku agenda presensi sehingga rentan terjadi kerusakan pada buku. Permasalahan lainnya adalah kesalahan dalam proses rekap presensi yang menghambat pembuatan laporan setiap bulan dalam satu periode sebagai bahan laporan akhir tahun dan sebagai data pendukung pengambilan keputusan. Penelitian ini berfokus pada perancangan sistem informasi presensi guru dan staf berbasis web dengan menggunakan bahasa pemrograman PHP dan basis data MySQL untuk membantu memantau kehadiran guru dan staf serta dapat menentukan kinerja guru dan staf dari hasil laporan presensi. (Triyono et al., 2018)

Penelitian sistem presensi lainnya yang pernah dilakukan sebelumnya berjudul “Perancangan Sistem Informasi Absensi Pegawai berbasis Web dengan Metode Waterfall (Studi Kasus: Kantor DBPR Tangerang Selatan)”. Permasalahan yang diangkat dalam penelitian ini adalah sistem pengambilan presensi pegawai saat ini masih menggunakan mesin sidik jari namun sistem presensi tersebut sering mengalami *error* pada saat pegawai melakukan presensi pulang sehingga pegawai harus melakukan presensi pulang dengan mencatat secara manual di buku presensi pegawai. Penelitian ini berfokus pada sistem informasi presensi pegawai berbasis web dengan metode *waterfall* untuk memudahkan proses presensi, pencarian data, meminimalisir kehilangan dan kesalahan pencatatan data di kantor DBPR Tangerang Selatan. (Olindo & Syaripudin, 2022)

Perbedaan penelitian yang sudah dilakukan dengan penelitian yang akan dilakukan oleh penulis adalah sistem presensi penulis berfokus pada jaringan intranet berbasis web sehingga karyawan hanya dapat melakukan presensi di dalam kantor, sedangkan sistem presensi pada penelitian di kantor DBPR Tangerang Selatan memungkinkan karyawan melakukan presensi di mana saja yang tidak sesuai dengan kebutuhan CV Trustline Indonesia.