

BAB 1. PENDAHULUAN

1.1 Latar Belakang

PT Pupuk Sriwidjaja Palembang (Pusri) dikenal sebagai pelopor produsen pupuk urea pertama di Indonesia sekaligus perusahaan induk (*holding company*) dengan 6 (enam) anak perusahaan. Dalam rangka menjalin hubungan dengan instansi pendidikan, PT Pusri menyediakan wadah bagi para calon penerus bangsa yakni pelajar, melalui program Kerja Praktik (KP) yang dapat membagikan pengalaman terjun langsung pada lingkungan kerja di PT Pusri (Pusri, 2021).

Pelaksanaan program KP di PT Pusri mendukung pelajar (siswa/i SMK dan mahasiswa/i) tingkat akhir yang terdaftar sebagai Peserta KP untuk meningkatkan *skill* terapan. *Skill* ini berupa penerapan ilmu dan pengetahuan yang didapat dari bangku sekolah atau kuliah melalui pengalaman bekerja langsung di unit kerja sesuai dengan kompetensi atau latar belakang pendidikan. Setelah memenuhi tugas yang diberikan selama pelaksanaan KP, PT Pusri akan memberikan honorarium bagi setiap Peserta KP sesuai dengan keaktifan dan jenjang pendidikan.

Terkait dengan pelaksanaan program KP tersebut, PT Pusri membagi peranan kepada 2 (dua) bagian pada Direktorat Keuangan dan Umum, yaitu Kompartemen Sumber Daya Manusia pada bagian Departemen Pembelajaran dan Pengembangan Sumber Daya Manusia (PPSDM) yang bertugas sebagai pengelola utama mulai dari proses penerimaan (pendaftaran calon Peserta KP), pembekalan, hingga penerbitan Surat Keterangan (SK) tanda ketuntasan, dan pengelolaan honorarium Peserta KP, serta Kompartemen Administrasi Keuangan pada bagian Departemen Akuntansi yang melanjutkan proses penyaluran dana honorarium kepada Peserta KP.

Proses pengelolaan honorarium Peserta KP pada PT Pusri dilakukan secara bertahap, dari Admin KP pada Departemen PPSDM yang mengelola data honorarium Peserta KP untuk kemudian dihitung berdasarkan kehadiran Peserta KP dalam kegiatan KP dengan menggunakan *Microsoft Excel*. Setelah itu, Admin KP akan menyerahkan laporan rekapitulasi honorarium Peserta KP kepada *Vice President* (VP) pada Departemen PPSDM, dan pihak *Finance* pada Departemen Akuntansi untuk kemudian memproses penyaluran honorarium ke Peserta KP.

Pengelolaan data honorarium Peserta KP di PT Pusri saat ini dilakukan menggunakan *Microsoft Excel*. Pengisian data kehadiran dilakukan secara tertulis

oleh Peserta KP. Lembar kehadiran Peserta KP ini digunakan Admin KP untuk mengelola honorarium dengan menginputkan ulang data tersebut ke dalam *Microsoft Excel*. Selain tidak cukup efisien, proses ini menghabiskan cukup banyak sumber daya dalam prosesnya. Selain itu, pengontrolan oleh VP PPSDM dan pendistribusian informasi dalam bentuk laporan dan slip pembayaran honorarium Peserta KP melewati alur yang cukup panjang untuk sampai ke tangan VP PPSDM dan *Finance* Akuntansi. Sistem ini juga belum menjamin integritas data dalam hal konsistensi dan keakuratan data honorarium, serta masih cukup rentan akan serangan virus, yang mana resiko kehilangan data pun cenderung tinggi. Selain itu juga, kesalahan yang ditemui dalam proses penginputan tersebut dapat beresiko pada kesalahan pembayaran, yang mana hal ini dapat mempengaruhi untung-rugi perusahaan hingga mempertaruhkan nama baik PT Pusri itu sendiri.

Berdasarkan uraian di atas, penulis mengambil alternatif penyelesaian dengan tugas akhir berjudul “Rancang Bangun Aplikasi Pengelolaan Honorarium Peserta Kerja Praktik (KP) pada PT Pusri Palembang (Pusri) Berbasis *Website*”.

Manfaat dari tugas akhir ini ialah terbangunnya aplikasi dengan data dan informasi yang dapat dikelola dan dikontrol secara *realtime*, tidak hanya dapat diakses melalui *file* yang tertaut di suatu perangkat. Pengelolaan data dapat dilakukan secara terstruktur, dimana awalnya penginputan data dilakukan sekaligus pada 1 (satu) *sheet* yang sama dengan kemungkinan terjadi kesalahan penginputan, dengan menu yang sudah terprogram di aplikasi data dapat diinputkan satu-persatu pada masing-masing Peserta KP tanpa khawatir mengganggu data peserta lainnya.

Aplikasi ini membantu proses pengelolaan komputerisasi honorarium dalam hal menginput, menghitung, dan mendistribusikan laporan dan slip pembayaran honorarium dengan automisasi dalam pembuatannya yang terintegrasi dengan data honorarium Peserta KP yang tersimpan di *database* dan dapat dicetak secara otomatis oleh VP PPSDM dan *Finance* Akuntansi, serta mempermudah Admin KP dan VP PPSDM dalam mengakses kembali riwayat data honorarium.

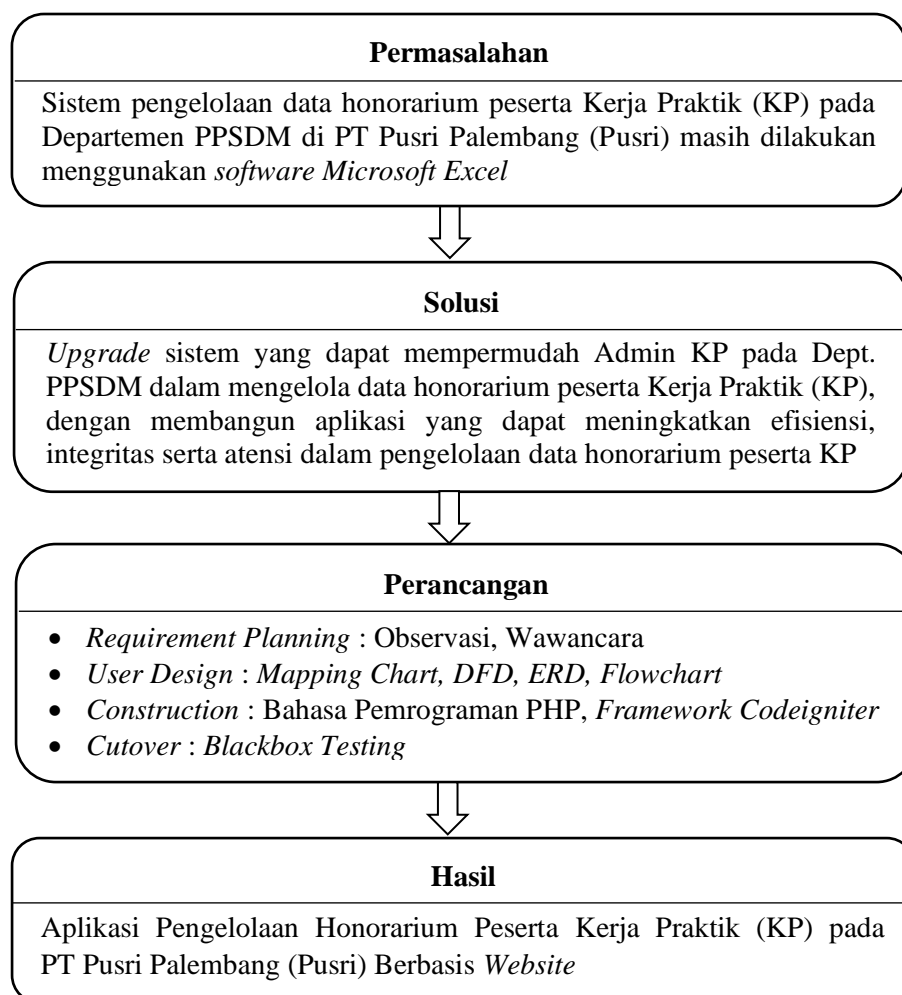
Dalam hal keamanan, konsistensi dan keakuratan data dapat terjaga dengan mempertahankan integritas seluruh data yang ada di dalam aplikasi dan *database* melalui pembatasan terkontrol pada tipe dan panjang data, serta data relasional yang didukung oleh *database MySQL*.

1.2 Tujuan

Berdasarkan masalah yang ditemukan, dapat ditentukan tujuan dari pelaksanaan Tugas Akhir ini ialah untuk membangun aplikasi yang memudahkan proses pengelolaan honorarium Peserta KP pada PT Pusri melalui Aplikasi Pengelolaan Honorarium Peserta KP pada PT Pusri Berbasis *Website*.

1.3 Kerangka Pemikiran

Dengan menilik permasalahan di atas, telah dirumuskan poin-poin penting. Berikut adalah kerangka pemikiran yang tersaji pada Gambar 1.



Gambar 1. Kerangka Pemikiran

1.4 Kontribusi

Melalui Tugas Akhir ini, penulis berharap untuk dapat menyalurkan kontribusi bagi beberapa pihak, yang di antaranya sebagai berikut :

- 1) Departemen PPSDM PT Pusri
 - a. Membantu kegiatan pengelolaan honorarium Peserta KP yang ditangani oleh Admin KP pada Departemen PPSDM di PT Pusri.
 - b. Memudahkan proses pengontrolan atau evaluasi kegiatan pengelolaan data honorarium KP oleh VP pada Departemen PPSDM di PT Pusri.
 - c. Memudahkan proses pendistribusian informasi dari Departemen PPSDM ke bagian *Finance* pada Departemen Akuntansi.
- 2) Departemen Akuntansi PT Pusri
 - a) Memudahkan *Finance* dalam menerima informasi slip pembayaran honorarium Peserta KP.
 - b) Membantu proses penyaluran honorarium ke Peserta KP dengan lebih cepat.
- 3) Peserta Kerja Praktik (KP) PT Pusri
 - a) Memudahkan Peserta KP dalam mengirimkan kehadiran untuk proses pengelolaan honorarium.
 - b) Memperoleh honorarium lebih cepat.
- 4) Politeknik Negeri Lampung
 - a. Menyalurkan ilmu pengetahuan baru berbentuk karya, sehingga dapat dijadikan referensi bagi generasi berikutnya.
 - b. Menciptakan hubungan yang baik dan harmonis dalam hal Pendidikan antara PT Pusri dan Politeknik Negeri Lampung.

BAB II. TINJAUAN PUSTAKA

2.1 Honorarium

Istilah honorarium tidak jarang terdengar di dunia kerja. Hal ini sering dikaitkan dengan proses keuangan yang termasuk ke dalam proses kegiatan kerja. Adapun definisi serta fungsi dan tujuannya di antaranya sebagai berikut.

2.1.1 Definisi Honorarium

Menurut Dekas (2019), honorarium merupakan upah atau kompensasi yang diberikan berdasarkan kinerja yang optimal dan hasil kerja yang dicapai sesuai standar perusahaan dalam kurun waktu tertentu. Dalam (Wikiapbn, 2015) juga tertulis bahwa honorarium ialah kegiatan pemberian bayaran atas penuntasan kegiatan tertentu. Dipetik dari buku “Kumpulan Catatan, Pengaturan-Pengaturan, serta Pelaksanaan Remunerasi pada Badan Layanan Umum” dari Budi Irianto, honorarium diberikan kepada individu yang memberikan layanan non-profesional atau tidak terikat kontrak kerja.

2.1.2 Mekanisme Honorarium

Berdasarkan Kamus Besar Bahasa Indonesia (KBBI), honorarium ialah suatu imbalan tanda jasa atau upah di luar gaji. Dekas (2019) menyatakan bahwa dalam penyalurannya, honorarium ini dibedakan menjadi 2 (dua) mekanisme, berikut.

1. Honorarium dalam Belanja Pegawai

Dalam hal ini, honorarium ialah istilah untuk nominal uang yang diberikan kepada pegawai honorer berupa tunjangan jasa di luar tugas pokok untuk mendukung tugas pokok serta fungsi suatu instansi atau organisasi.

2. Honorarium dalam Belanja Non-Pegawai

Honorarium jenis ini dibagi menjadi dua bagian, yaitu:

- a. Honor *Output* Kegiatan, ialah honor yang dibayarkan secara tidak tetap kepada pekerja yang ditugaskan untuk kegiatan insidental.
- b. Honor Operasional Satuan Kerja, ialah honor dengan pembayaran tidak tetap yang diberikan untuk kegiatan yang operasional secara berkelanjutan dari awal hingga akhir tahun anggaran.

2.2 Rancang Bangun Aplikasi

Dalam melakukan rancang bangun aplikasi honorarium berbasis *Website*, diperlukan beberapa *resources* atau sumber daya agar dapat membangun aplikasi sesuai dengan yang diharapkan. Adapun sumber daya yang dipakai dalam perancangan aplikasi ini antara lain adalah sebagai berikut.

2.2.1 Rancang Bangun

Dalam tulisannya, Hidayat, dkk., (2022) menyatakan bahwa perancangan adalah serangkaian prosedur untuk perencanaan mulai dari pemikiran atau gagasan, serta melibatkan proses penyelesaian yang mencakup penerjemahan hasil analisis sistem menjadi bahasa pemrograman, hal-hal yang sudah teratur tersebut dapat digunakan dan berfungsi dengan baik.

Adapun pembangunan sistem aplikasi, menurut Girsang (2019) dapat diartikan sebagai kegiatan mengembangkan kinerja rancangan sistem yang ingin dipenuhi dan diraih dengan penerapan bahasa pemrograman tertentu, pembangunan sistem baru atau penyempurnaan seluruh atau sebagian sistem yang sudah ada.

Sehingga rancang bangun dapat didefinisikan sebagai proses pembangunan atau penciptaan suatu implementasi sistem yang belum ada dalam suatu instansi dengan menerjemahkan hasil analisis ke dalam bentuk aplikasi, yang kemudian akan dibangun sistem baru atau penyempurnaan sistem yang sudah ada.

2.2.2 Aplikasi

Di samping keberadaan perangkat keras (*hardware*) dan jaringan (*network*), sistem komputer tidak akan sempurna tanpa adanya aplikasi atau perangkat lunak (*software*). Aplikasi merupakan sistem terkomputerisasi yang digunakan manusia dengan sarana komputer yang kegunaannya ialah menjalankan perintah dari pengguna aplikasi dengan tujuan untuk memperoleh hasil yang lebih tepat sesuai dengan tujuan pembuatan aplikasi.

Aplikasi ini merupakan salah satu solusi dengan menggunakan salah satu teknik untuk memproses data aplikasi yang biasanya berfokus pada perhitungan atau pemrosesan data yang diinginkan atau diharapkan. Aplikasi disusun oleh prosedur penggunaan instruksi atau instruksi yang disiapkan dalam komputer sehingga komputer dapat memproses masukan (*input*) dan kemudian mengubahnya

menjadi keluaran (*output*). (Yono, 2021). Aplikasi dibagi menjadi beberapa jenis, yaitu aplikasi berbasis *desktop*, aplikasi berbasis *mobile*, aplikasi berbasis *android*, dan aplikasi berbasis *Website*.

2.2.3 Aplikasi Website

Website atau situs web adalah halaman informasi yang tersedia di internet. Adapun aplikasi berbasis *Website* telah banyak digunakan oleh banyak instansi untuk menjadi bagian dari strategi teknologi informasi dan administrasinya. Hal ini didasari oleh keunggulan dari aplikasi *Website* itu sendiri seperti yang dikutip dari (Solihin, 2016) yaitu di antaranya sebagai berikut:

- a. Tidak memerlukan proses instalasi, dimana hal ini memudahkan pengguna dalam menggunakan aplikasi tanpa perlu spesifikasi komputer yang besar, dan proses pembaharuan yang terjadi di aplikasi dalam hal *update* kinerja maupun fitur tidak akan langsung mempengaruhi pengguna.
- b. Mampu diakses dari mana saja melalui jaringan, sesuai sifatnya yang *multi platform*, sehingga dapat menyajikan informasi dari *web browser* pada berbagai jenis sistem operasi.
- c. Penyimpanan data dilakukan pada sisi server, dimana proses pengaksesan data pada sisi pengguna dapat disesuaikan dengan kebutuhan.
- d. Pengguna relatif aman dari virus, karena aplikasi dijalankan di *browser*.

2.2.4 Hypertext Preprocessor (PHP)

PHP ialah bahasa pemrograman yang umum dipakai dalam pengembangan aplikasi berbasis web dinamis. PHP atau *Hypertext Preprocessor* merupakan sebuah bahasa pemrograman *open source* yang berjalan pada *server side scripting*.

Alur kerja PHP diawali dari permintaan pada halaman web oleh *browser*. Lalu *browser* mendapatkan alamat dari pada *web server* melalui URL atau alamat web, mengidentifikasi alamat, serta menyalurkan informasi yang diperlukan *web server* untuk diteruskan ke mesin pengoperasi PHP yang memproses dan memberi hasil berupa kode HTML ke bagian *web server* untuk disampaikan ke *client* (Krisbiantoro & Abda'u, 2021).

Dalam pengoperasiannya, terdapat beberapa fungsi dari PHP yang di antaranya ialah sebagai berikut:

- a. Membuat konten dinamis yang lebih menarik bagi pengguna, dengan informasi yang dapat berubah-ubah sesuai kondisi tertentu.
- b. Memodifikasi data dalam *database*, dengan CRUD untuk mengolah data.
- c. Memiliki kontrol atas *file server* yang beroperasi di *server* melalui prosedur CRUD untuk mengolah *database*.
- d. Membuat dan mengumpulkan data melalui formulir yang dibuat dengan dasar HTML, dan dilengkapi dengan fungsi PHP untuk mendapatkan data pengguna dan menampilkannya kembali ke sisi pengguna.
- e. Mengatur akses pengguna, dengan batasan yang dibuat oleh fungsi PHP.
- f. Mengenkripsi data yang sensitif bagi pengguna yang dikenal dengan kode samaran unik. Fungsi ini dikenal dengan istilah *hashing*.

2.2.5 My Structured Query Language (MySQL)

MySQL adalah *database management system* dengan akses *open source* dan disusun oleh perintah dasar dari SQL (*Structured Query Language*) berjenis *Relational Database Management System* (RDBMS) dengan bahasa SQL yang menghubungkan aplikasi dengan *database server* dengan struktur tabel yang fleksibel, tipe data bervariasi, mendukung *multi-user*, dan mendukung integrasi dengan bahasa pemrograman lain, serta keamanan yang terjamin (Yasin, 2022).

Perangkat lunak *MySQL* mencakup *server MySQL*, dan program utilitas yang mendukung administrasi *database MySQL*, serta beberapa perangkat lunak pendukung. Pusat kendali dari sistem adalah *server MySQL* yang merupakan sistem *database* yang menangani semua instruksi *database* (Silalahi, 2022).

2.2.6 PHPMYAdmin

PhpMyAdmin merupakan aplikasi bersifat *open source* dengan fungsi untuk mempermudah pengelolaan *MySQL* berjalan di banyak sistem operasi, selama mengaplikasikan web server dan *MySQL*. *Phpmyadmin* mampu membuat *database*, membuat tabel, menambahkan, memperbarui, mengubah, menghapus, mengimpor, dan mengekspor tabel *database MySQL* menggunakan GUI dan hal itu

dapat mempermudah *developer*, tanpa harus mengetikkan perintah SQL yang banyak secara manual (Sofwan, 2017).

2.2.7 CodeIgniter (CI)

CodeIgniter atau CI adalah salah satu *framework* PHP berlisensi bebas dengan arsitektur MVP. Konsep MVP (*Model, View, Controller*) dapat memisahkan kode program dengan pengelolaan *database*. Dimana *model* berisi perintah untuk mengirimkan data ke bagian *controller* dan berhubungan langsung pada *database* dengan fungsi merepresentasikan data, mengelola respon terhadap permintaan, dan menyediakan hak akses untuk manipulasi data dengan mengambil data dari *database* lalu memasukkannya ke dalam *database*. *View* mengatur data yang diambil dari *controller* untuk ditampilkan ke sisi pengguna. *Controller* berfungsi untuk meneruskan suatu perintah ke bagian *model* untuk selanjutnya memperoleh data yang diinginkan, kemudian mengirimkannya ke bagian *view* agar tampil ke sisi pengguna (Hidayat, 2013).

CI dikenal dengan kemudahannya untuk digunakan oleh pemula (Rofiah S., 2018). Dengan ukurannya yang kecil, CI mendukung *developer* pemula dengan perangkat yang berkapasitas kecil. Selain ringan, CI memiliki performa yang cepat (Prasena, dkk., 2020). Dalam segi keamanan, CI juga menjamin *Website* dari serangan injeksi SQL (Purbo, 2021), yang merupakan aksi peretasan dengan mengubah perintah SQL dalam memori aplikasi klien yang berpotensi merusak sistem basis data dengan mengganti data yang ada (Pratama, dkk., 2022).

2.2.8 Rapid Application Development (RAD)

Rapid Application Development (RAD) merupakan bagian dari *System Development Life Cycle* (SDLC) dengan pengembangan sistem aplikasi yang mencakup suatu metode pengembangan berorientasi objek (Pricilla & Zulfachmi, 2021). Metode pengembangan ini memusatkan pada siklus waktu pengembangan yang relatif singkat dalam pengembangan aplikasi. pada prosesnya, metode ini memiliki beberapa tahap yaitu *Requirements Planning* (Rancangan Kebutuhan), *User Design* (Desain Sistem Aplikasi), *Construction* (Pembangunan Aplikasi), dan *Cutover* (Pengujian Aplikasi) (Corputy, dkk., 2018).

Pada tahapan implementasi RAD, calon pengguna menjadi bagian dari proses pengembangan, dan turut mengambil keputusan pada tahapan pengembangan sistem. Dengan metode RAD, akan dapat dengan cepat dihasilkan, dikarenakan aplikasi melalui proses rancangan berdasarkan kriteria dari para calon pengguna. Hal tersebut mampu meminimalisir waktu setelah tahap implementasi untuk pengembangan ulang, meskipun *developer* dapat kapanpun kembali ke tahap atau komponen sebelumnya. Sehingga apabila normalnya pengembangan aplikasi dibutuhkan waktu lebih dari 180 hari, namun dengan metode RAD suatu aplikasi dapat dikembangkan dalam kurun waktu 30-90 hari (Noertjahyana, 2012).

2.2.9 Visual Studio Code

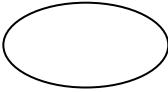
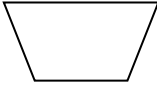


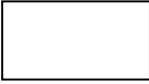


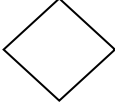

Visual Studio Code merupakan *text editor* dengan pengoperasian di semua *desktop (multi-platform)*, serta komplit dan handal dalam mendukung banyak bahasa pemrograman. Meskipun gratis, (Tan, dkk, 2022) menjelaskan bahwa *visual studio code* memiliki banyak fitur yang di antaranya adalah sebagai berikut:

1. *Basic Editing*, fungsi utama yang lengkap di kelasnya, dengan berbagai *formatting code*, *auto-save*, dan *hotkey* agar *editing* nyaman, cepat, praktis.
2. *Debugging*, yang dapat memudahkan pengguna dalam mengelola program yang dibangun. Dalam hal ini termasuk memodifikasi, menyusun, dan menjalankan kode pemrograman beberapa kali dalam waktu yang singkat.
3. *Extension Marketplace*, memungkinkan pengguna untuk menjelajahi dan menambah komponen selain di aplikasi, dengan ekstensi lain dengan cepat.
4. *IntelliSense*, yang terintegrasi secara otomatis dan berfungsi untuk menampilkan *sugesti possible code* berdasarkan bahasa pemrograman atau *database* yang digunakan.

2.2.10 Mapping Chart

Mapping chart adalah diagram yang menginterpretasikan rincian dari setiap alur proses aliran informasi di dalam sistem dari awal hingga selesai, serta bagian-bagian yang terlibat di dalamnya (Rozak, dkk., 2018). Simbol yang digunakan pada *mapping chart* di antaranya tertera pada Tabel 1.

Tabel 1. Simbol-simbol *Mapping Chart*

Simbol	Keterangan
	<i>Simbol Terminator</i> (Simbol yang menggambarkan awal/akhir suatu program)
	<i>Simbol Manual Activity</i> (Simbol untuk proses <i>input</i> atau <i>output</i> secara manual)
	<i>Simbol Database</i> (Simbol gambaran dari penyimpanan data terkomputerisasi)
	<i>Simbol Connector</i> (Simbol keluar/masuk proses yang bersambung pada halaman sama)
	<i>Simbol Process</i> (Simbol untuk kondisi dari fungsi yang menghasilkan suatu aksi)
	<i>Simbol Document</i> (Simbol untuk <i>output</i> informasi dalam bentuk dokumen jadi)
	<i>Simbol Display</i> (Simbol yang menggambarkan alat <i>output</i> yang dipakai)
	<i>Simbol Decision</i> (Simbol yang menggambarkan kondisi dengan pilihan ya atau tidak)
	<i>Simbol Flow Line</i> (Simbol arah aliran program)




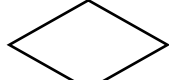
Sumber : (Verawati, & Liksha, P., D., 2021)

2.2.11 Entity Relationship Diagram (ERD)

ERD atau *Entity Relationship Diagram* ialah pemodelan basis data yang menjadi acuan dalam menggambarkan data untuk dikelola bersamaan dengan rancangan aplikasi yang dibangun (Li & Chen, 2009).

ERD menyimpan data dalam tabel beserta *field* yang ada di dalamnya, yang digambarkan dengan entitas dan atribut, serta relasi di antaranya. Berikut penjelasan simbol-simbol pada ERD disajikan dalam Tabel 3.

Tabel 2. Simbol-simbol *Entity Relationship Diagram* (ERD)

Simbol	Keterangan
	<i>Simbol Alur</i> (Simbol yang menghubungkan relasi antar entitas)
	<i>Simbol Entitas</i> (Simbol yang menjelaskan entitas sebagai pembeda antar objek)
	<i>Simbol Atribut</i> (Simbol yang menggambarkan suatu atribut dari suatu entitas)
	<i>Simbol Relasi</i> (Simbol yang berarti relasi antar entitas satu dan yang lainnya)

Sumber : (Solikin, dkk., 2018)

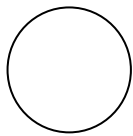
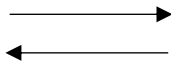


2.2.12 *Data Flow Diagram* (DFD)

DFD atau *Data Flow Diagram* dapat diketahui sebagai diagram alir yang merepresentasikan aliran data yang diolah dalam proses yang berjalan pada suatu sistem, dari awal proses *input* hingga ke *output* (Rozak, dkk., 2018). Adapun dalam pemodelannya, DFD dibagi dalam 3 bentuk, yaitu sebagai berikut.

- DFD Level 0, ialah diagram paling dasar yang meliputi gambaran interaksi pada sistem dengan entitas eksternal.
- DFD Level 1, yaitu diagram yang merupakan penguraian dari DFD level 0, dimana informasi dibagi menjadi unit yang lebih sederhana untuk dipahami.
- DFD Level 2, yaitu diagram yang memuat informasi lebih detail terkait proses yang terjadi pada DFD level 1. DFD level 2 ini hanya dimiliki oleh sistem tertentu.

Proses pada DFD digambarkan dalam simbol. Simbol-simbol yang digunakan pada DFD yang di antaranya akan disajikan pada Tabel 4.

Tabel 3. Simbol-simbol *Data Flow Diagram* (DFD)

Simbol	Keterangan
	<i>Simbol Process</i> (Simbol sebagai gambaran proses dari <i>input</i> menjadi <i>output</i>)
	<i>Simbol Data Flow</i> (Simbol dari aliran suatu data dari <i>input</i> ke <i>output</i>)
	<i>Simbol Storage</i> (Simbol gambaran dari penyimpanan pada <i>database</i>)
	<i>Simbol External Entity</i> (Simbol dari entitas yang saling berkomunikasi pada sistem)

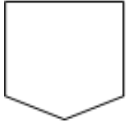
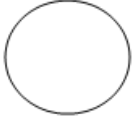


Sumber : (Budiman, dkk., 2021)

2.2.13 *Flowchart*

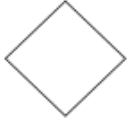


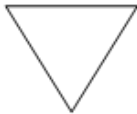




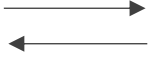
Menurut Sahid (2010), *Flowchart* dapat diartikan sebagai diagram yang secara logis mewakili aliran suatu program atau proses sistem. Diagram alur terutama digunakan sebagai alat komunikasi dan dokumentasi.

Dalam pembuatan *flowchart* menggunakan simbol-simbol yang menggambarkan aliran informasi di dalam sistem. Simbol-simbol *flowchart* yang digunakan biasanya antara lain tertera pada Tabel 5.

Tabel 4. Simbol-simbol *Flowchart*

Simbol	Keterangan
	<i>Simbol Off-line Connector</i> (Simbol yang menggambarkan proses keluar/masuknya data pada lembar/halaman lain)
	<i>Simbol Connector</i> (Simbol keluar/masuknya proses yang bersambung pada lembar/halaman)
	<i>Simbol Process</i> (Simbol yang menggambarkan suatu proses yang dilakukan oleh komputer)
	<i>Simbol Manual Operation</i> (Simbol dari pengolahan data yang tidak dilakukan komputer)

Tabel 5. Lanjutan Tabel Simbol-simbol *Flowchart*

Simbol	Keterangan
	<i>Simbol Decision</i> (Simbol yang menunjukkan kondisi dengan beberapa pilihan aksi)
	<i>Simbol Predefined Process</i> (Simbol dari persiapan penyimpanan pengolahan)
	<i>Simbol Terminal</i> (Simbol untuk suatu permulaan atau akhir dari suatu program)
	<i>Simbol Off-line Storage</i> (Simbol yang menunjukkan bahwa data akan disimpan)
	<i>Simbol Punched Card</i> (Simbol <i>input</i> yang berasal dari kartu atau <i>output</i> yang ditulis ke kartu)
	<i>Simbol Input-Output</i> (Simbol jenis alat yang dipakai dalam media <i>input</i> dan <i>output</i> data)
	<i>Simbol Disk and On-Line Storage</i> (Simbol <i>input</i> yang berasal dari <i>disk/output</i> dikirim ke <i>disk</i>)
	<i>Simbol Display</i> (Simbol yang menyatakan peralatan <i>output</i> yang digunakan yaitu layar, <i>plotter</i> , <i>printer</i> , dan sebagainya)
	<i>Simbol Flow Line</i> (Simbol yang menyatakan aliran suatu proses)

Sumber : (Khesya, 2021)

2.3 Jurnal Terkait

Terdapat beberapa teori yang menjadi acuan dalam penyusunan tugas akhir ini, yang diambil dari berbagai sumber riset. Berikut karya ilmiah yang dijadikan sebagai referensi dalam penyelesaian tugas akhir ini.

1. Puspitasari (2014) yang terbit dengan judul “Aplikasi *CI-Smart* Sebagai Sistem Informasi Untuk Mengelola Honor Dosen dengan Pendekatan Berorientasi Objek Studi Kasus: STMIK AMIKOM Yogyakarta”. Artikel ini dapat menjadi referensi aplikasi dengan *framework* berupa *CodeIgniter* yang

mampu menghasilkan aplikasi ringan dengan akses yang cepat, selain itu terdapat *Smarty* yang merupakan *template engine* yang handal dan membuat web lebih tertata rapi dan menarik.

2. Yakub, M., R. & Solihin H., H. (2019) yang berjudul “Sistem Informasi Manajemen Honorarium SMAN 1 Blanakan”. Artikel ini memberikan referensi untuk membangun proses perhitungan yang terstruktur, dan minim kesalahan. Aplikasi ini dibangun dengan metode pengembangan *waterfall*, dengan metode pengumpulan data melalui proses observasi dan wawancara.
3. Gunardi & S. Ikhsan (2020) yang diterbitkan dengan judul “Sistem Informasi Honorarium Berbasis Web di Dinas Pemakaman dan Pertamanan Kota Bandung”. Artikel ini dapat menjadi pedoman dalam perancangan aplikasi honorarium dengan prosedur pembayaran sesuai standard, dan minim masalah dalam pembuatan laporan honorarium.
4. Crystiana (2019) yang berjudul “Aplikasi Perhitungan dan Perincian Gaji Berbasis Web pada Pusbang Wakaf Daarut Tauhiid Bandung Menggunakan *Framework CodeIgniter*”. Artikel ini dapat memberikan alternatif bagi *Finance* dan SDI untuk mengelola tunjangan, dan penggajian dan potongnya secara lebih mudah dan efisien.
5. Liyana, K. F., Zuriati, & Maulini, Rima (2011) yang terbit dengan judul “Aplikasi *Payroll* Berbasis Web pada Yayasan Citra Angkasa Tercipta (CAT)”. Artikel ini menggunakan metode pengembangan *prototype* dalam pembuatan aplikasi dengan sistem *database* untuk menghasilkan aplikasi yang lebih mudah dan cepat diakses, serta penyimpanan data yang dapat tersusun dengan lebih terstruktur, aman dan mudah diakses.