

I. PENDAHULUAN

1.1 Latar Belakang

PT Telekomunikasi Indonesia merupakan suatu Badan Usaha Milik Negara yang bergerak di bidang jasa layanan teknologi informasi dan komunikasi (TIK) dan jaringan telekomunikasi di Indonesia. Pemilik saham terbesar pada perusahaan ini dipegang oleh pemerintah Indonesia, yakni sebesar 52.09%, dan sisanya dipegang oleh publik. Saham perusahaan ini diperdagangkan di Bursa Efek Indonesia (BEI) dengan kode “TLKM” dan *New York Stock Exchange* (NYSE) dengan kode “TLK”.

Perusahaan ini tentunya memiliki beberapa divisi bagian atau bidang, seperti bidang pemesanan yang disebut dengan IOC-W (*Integrated Operator Center – WITEL*), *Corporate Customer Access Network (CCAN)* untuk menjamin kualitas layanan yang prima ke pelanggan dari segmen *Enterprise*, pemerintahan dan bisnis, *Access Service Operation (ASO)* untuk melakukan perbaikan gangguan yang terjadi setelah pemasangan layanan *indihome*, *War Room*, *Access Maintenance Operation (AMO)*, *Network Area (NETA)*, *Wholesale Access Network (WAN)*, *Consumer Service (CS)*, *Customer Care (CC)*, *Business Government Enterprise Service (BGES)*, *Logistic & General Support (LOG & GS)*, *Human Resource & Community Development Center (HR & CDC)*, *Payment Collection (PC)*, dan juga *Junior Operation Manager (JOM)* *Information Technology Support (IT SUPP)*.

Pada divisi ASO (*Access Service Operation*), terdapat beberapa staff dan karyawan yang memegang jabatan, seperti *Human Development (HD)*. Salah satu tugasnya adalah untuk memvalidasi status *order* untuk memasang jaringan dari

customer dan memasukkan datanya ke dalam *Google Documents (G-Docs)*. Status *order* berupa PS (Pasang) atau Belum PS (Belum Pasang).

Semua proses validasi masih dilakukan secara manual, masih sangat bergantung dengan keberadaan *HD* dan para staff yang lainnya. Semua data disimpan dalam satu basis data, kemudian diolah dan dikelola oleh *HD* untuk informasi terbaru terkait dengan status *order* tersebut. Data tersebut menjadi semakin banyak, menjadi sebuah kumpulan data besar dan data tersebut tidak diolah dengan baik, karena tidak adanya sebuah aplikasi atau *tools* untuk mempermudah hal tersebut. Selain itu, karena terjadi penumpukan data, maka staff yang bekerja pun akhirnya malas untuk mengecek data tersebut dan bekerja dengan seadanya saja. Permasalahannya adalah pihak *management* ingin mengukur performansi totalitas (*performance*). Apabila pihak *sales* atau *management* ingin mengetahui status *order* tersebut, maka mereka masih harus berkoordinasi ke *HD* secara manual, atau bila tidak berkoordinasi kepada *HD* secara langsung, mereka berkoordinasi terlebih dahulu kepada atasan mereka, baru nantinya atasan mereka berkoordinasi kepada bagian *HD*. Dengan keberadaan seperti itu, maka hal yang menjadi kelemahan dari sistem yang lama ini adalah status *order* baru dapat diketahui h-1. Hal ini akan menjadi sulit ketika orang yang bekerja di bagian *Human Development* telah berganti divisi atau pindah bagian. Karena orang yang menggantikan tidak memegang kendali penuh terhadap pengelolaan data status *order* yang dimiliki oleh *HD* sebelumnya.

Basis data pada *Google Documents* tadi memang masih akan digunakan, tapi semuanya bergantung pada *HD* yang baru nantinya, apakah data tersebut akan rajin diperbarui atau tidak. Maka dari itu, aplikasi pencarian data ini akan dapat

mengatasi permasalahan yang ada, seperti aplikasi yang dapat menampung data yang besar sehingga menjadi sebuah pangkalan data, sangat memudahkan bagian *sales* dan *management* untuk melakukan pengecekan status *order* secara massal, aplikasi pun dapat mengefisiensi tenaga karena tidak harus berkoordinasi secara manual kepada *HD*, kemudian status *order* yang dapat diketahui secara *realtime*, tidak h-1 lagi. Aplikasi ini pun akan sangat ringan (*user friendly*) jadi sangat mudah untuk digunakan.

1.2 Tujuan

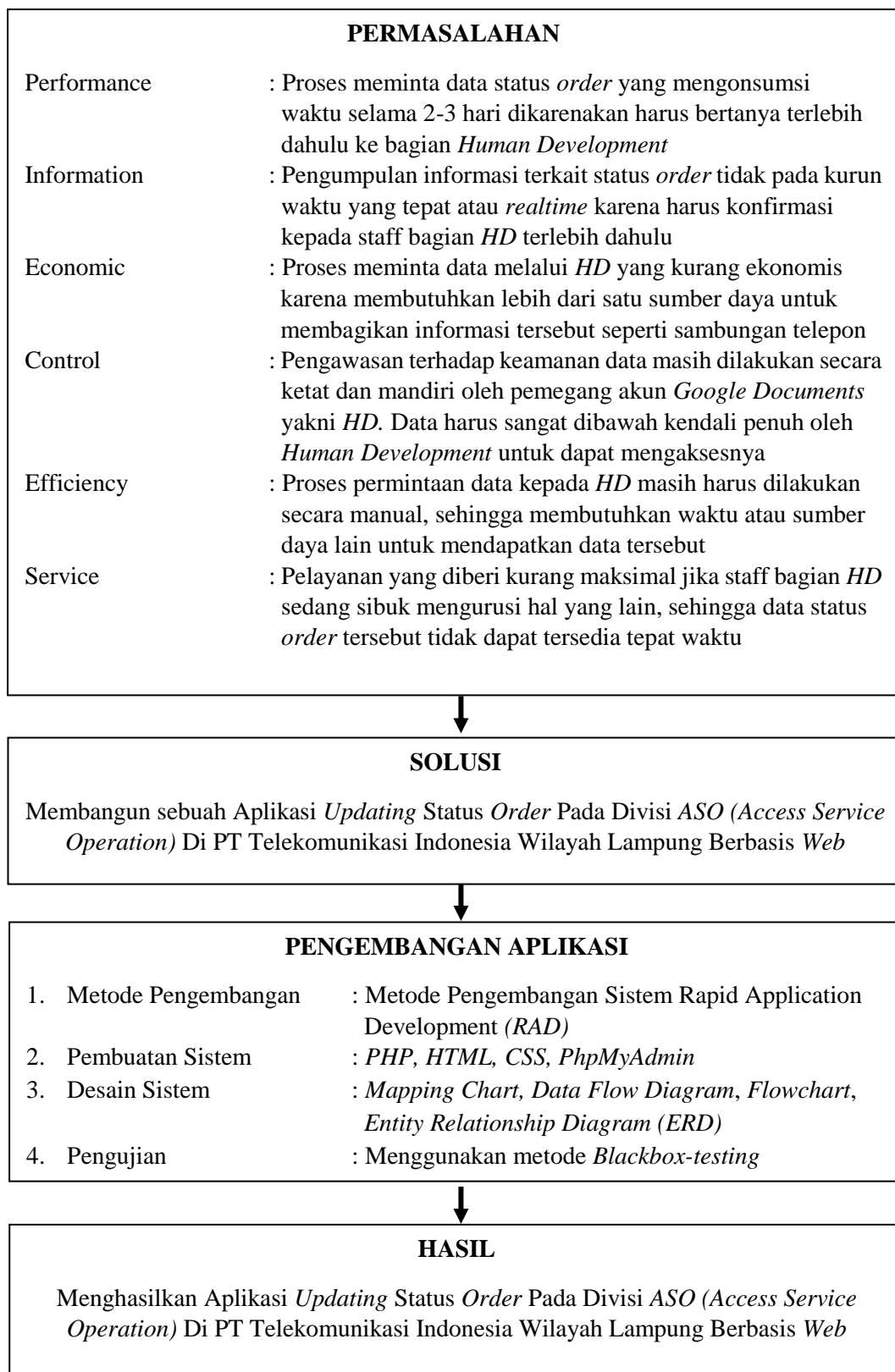
Tujuan dalam penyusunan tugas akhir ini adalah menghasilkan Aplikasi *Updating Status Order* Pada Divisi *ASO (Access Service Operation)* di PT Telekomunikasi Indonesia Wilayah Lampung Berbasis *Web*.

1.3 Kerangka Pemikiran

Validasi data merupakan hal yang penting untuk mengetahui status terbaru dari pemesanan, maka dari itu aplikasi ini sangatlah penting untuk mendapatkan informasi yang cepat dan tepat. Selama ini, sistem pengecekan validasi data di divisi *HD* PT Telkom Witel Lampung masih dilakukan secara manual.

Sistem dimulai ketika pihak *sales* atau *management* menanyakan tentang status *order* terbaru kepada bagian *Human Development* atau ketika mereka menanyakan ke atasan mereka, lalu atasan mereka menanyakan ke bagian *Human Development* pula, lalu staff yang bekerja di bagian *Human Development* pun mengecek terlebih dahulu data yang diminta, kemudian ketika berhasil ditemukan, baru *HD* memberitahu hasil terbaru dari data status *order* tersebut. Hal tersebut tentunya akan menggunakan waktu yang lebih lama untuk berbagi informasi.

Berdasarkan permasalahan tersebut, diperlukan suatu aplikasi yang dapat membangun kinerja bagian staff *Human Development* dalam membagikan hasil data terbaru terkait status pemesanan, sehingga tidak mengkonsumsi waktu yang lebih lama untuk berbagi informasi. Sistem dikembangkan dengan menggunakan metode *RAD (Rapid Application Development)*. Metode *RAD* dapat mempercepat dan mempersingkat waktu dalam pengembangan sistem. Metode *RAD* memiliki empat tahapan yang melibatkan pengguna dan penganalisis dalam tahapan perancangan, penilaian, dan penerapan aplikasi. Empat tahapan *RAD* yaitu *Requirements Planning*, *User Design*, *Construction* dan *Cutover*. Kerangka pemikiran disajikan dalam Gambar 1.



Gambar 1. Kerangka Pemikiran

1.4 Kontribusi

Kontribusi yang dapat diperoleh dari aplikasi yang akan dikembangkan:

1. Staff bagian *Human Development* pada divisi *ASO* di PT Telekomunikasi Indonesia dapat memudahkan dalam proses pembagian data berupa status *order* terbaru kepada pihak *sales* atau *management*.
2. Politeknik Negeri Lampung diharapkan dapat menjadi sarana pengetahuan baru dan menjadi referensi untuk lebih memahami aplikasi pencarian data yang sedang berjalan.
3. Penulis dapat meningkatkan ilmu pengetahuan dan keterampilan dalam menyajikan informasi dalam sebuah aplikasi.

II. TINJAUAN PUSTAKA

2.1 Aplikasi

Aplikasi merupakan penerapan untuk menyimpan sesuatu hal, data, permasalahan dan pekerjaan ke dalam suatu sarana atau media yang dapat digunakan untuk diterapkan menjadi sebuah bentuk yang baru. (Siregar, H. F., Siregar, Y. H., & Melani, 2018)

2.2 Pengolahan Data

Pengolahan data merupakan suatu proses menerima data sebagai masukan (*input*), memproses (*processing*) menggunakan proses tertentu, dan mengeluarkan hasil proses data tersebut dalam bentuk informasi (*output*). (Putra et al., 2019)

2.3 Website

Website merupakan sebuah tempat yang memungkinkan seseorang menyatakan dirinya, hobinya, pengetahuannya, produk yang dijualnya dan apapun juga yang dapat di akomodasikan oleh teks, tulisan, gambar, video, animasi dan *file* multimedia lainnya. (Muslim & Dayana, 2016)

2.4 Database

Database atau basis data merupakan kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk mendapatkan informasi dari basis data tersebut. (Andaru, 2018)

2.5 *Internet*

Internet merupakan komunikasi jaringan komunikasi global yang menghubungkan seluruh komputer di dunia meskipun berbeda sistem operasi dan mesin. (Nofyat et al., 2018)

2.6 *Hypertext Markup Language (HTML)*

Hypertext Markup Language (HTML) merupakan bahasa standar yang digunakan untuk menampilkan halaman *web*. (Novendri et al., 2019)

2.7 *PHP: Hypertext Preprocessor (PHP)*

PHP merupakan singkatan dari *PHP: Hypertext Preprocessor*. Saat pertama kali dikembangkan oleh programmer bernama Rasmus Lerdoff, *PHP* awalnya merupakan singkatan dari *Personal Home Page Tools*. (Sari, 2017)

2.8 *Visual Studio Code*

Visual Studio Code adalah *editor* kode sumber yang dikembangkan oleh *Microsoft* untuk *Windows*, *Linux* dan *macOS*. Ini termasuk dukungan untuk debugging, kontrol *git* yang tertanam dan *GitHub*, penyorotan sintaksis, penyelesaian kode cerdas, snippet, dan *refactoring* kode. (Muakhor, 2020)

2.9 *XAMPP*

XAMPP merupakan sebuah aplikasi yang dapat menjadikan komputer kita menjadi sebuah server. Kegunaan *XAMPP* ini untuk membuat jaringan lokal sendiri dalam artian dapat membuat *website* secara *offline* untuk masa coba-coba di komputer sendiri. (Josi, 2017)

2.10 *MySQL*

MySQL merupakan sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis. *MySQL* merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya *SQL* (*Structured Query Language*). *SQL* adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data. (Lestanti & Susana, 2016)

2.11 *MD5*

MD5 merupakan salah satu dari serangkaian algoritma (merupakan salah satu fungsi *Hash*) *message digest* yang didesain oleh Profesor Ronald Rivest dari MIT (Rivest, 1994). Saat kerja analitik menunjukkan bahwa pendahulu *MD5*, yaitu *MD4* mulai tidak aman, *MD5* kemudian didesain pada tahun 1991 sebagai pengganti dari *MD4* (kelemahan *MD4* ditemukan oleh Hans Dobbertin). Dikarenakan *MD5* hanya menggunakan satu langkah pada data, jika dua buah awalan dengan *hash* yang sama dapat dibangun, sebuah akhiran yang umum dapat ditambahkan pada keduanya untuk membuat kerusakan lebih masuk akal. Dan dikarenakan teknik penemuan kerusakan mengijinkan pendahuluan kondisi *hash* menjadi arbitari tertentu, sebuah kerusakan dapat ditemukan dengan awalan apapun. Proses tersebut memerlukan pembangkitan dua buah *file* perusak sebagai *file template*, dengan menggunakan blok 128-byte dari tatanan data pada 64-byte batasan, file-file tersebut dapat mengubah dengan bebas dengan menggunakan algoritma penemuan kerusakan. (Selsa, 2021)

2.12 *Framework*

Framework merupakan kumpulan intruksi-intruksi yang dikumpulkan dalam *class* dan *function-function* dengan fungsi masing-masing untuk memudahkan *developer* dalam memanggilnya tanpa harus menuliskan *syntax* program yang sama berulang-ulang serta dapat menghemat waktu. (Destiningrum & Adrian, 2017)

2.13 *CodeIgniter*

CodeIgniter merupakan sebuah *framework php* yang bersifat *open source* dan menggunakan metode *MVC (Model, View, Controller)* untuk memudahkan *developer* atau *programmer* dalam membangun sebuah aplikasi berbasis *web* tanpa harus membuatnya dari awal. (Sallaby & Kanedi, 2020)

2.14 *Object Oriented Programming (OOP)*

Object Oriented Programming merupakan sebuah pendekatan untuk pengembangan suatu perangkat lunak, dimana dalam struktur perangkat lunak tersebut didasarkan kepada interaksi objek dalam penyelesaian suatu proses atau tugas. (Fridayanthie & Charter, 2016)

2.15 *Model-View-Controller (MVC)*

Model-View-Controller (MVC) adalah sebuah konsep yang diperkenalkan oleh penemu Smalltalk (Trygve Reenskaug) untuk meng-enkapsulasi data bersama dengan pemrosesan (*model*), mengisolasi dari proses manipulasi (*controller*) dan tampilan (*view*) untuk direpresentasikan pada sebuah *user interface*. (Dayat & Angriani, 2017)

2.16 Rapid Application Development (RAD)

Rapid Application Development (RAD) atau *rapid prototyping* merupakan model proses pembangunan perangkat lunak yang tergolong dalam teknik inkremental (bertingkat). *Rapid Application Development (RAD)* menekankan pada siklus pembangunan pendek, singkat, dan cepat. Waktu yang singkat adalah batasan yang penting untuk model ini. *Rapid Application Development (RAD)* menggunakan metode iteratif (berulang) dalam mengembangkan sistem dimana model kerja sistem dikonstruksikan diawal tahap pengembangan dengan tujuan menetapkan kebutuhan pengguna. Model kerja digunakan hanya sesekali saja sebagai basis desain dan implementasi sistem akhir. (Sagala, 2018)

RAD memiliki empat tahapan, yakni:

2.16.1 Requirements Planning

Tahapan ini menganalisis permasalahan dari sistem yang ada, lalu melakukan identifikasi kebutuhan sistem.

2.16.2 User Design

Tahapan ini membuat desain dari sistem yang akan dibangun, yang kemudian diimplementasikannya. Desain yang akan dibuat untuk sistem yang baru berupa alur data sistem, relasi basis data, gambaran sistem dan *interface* atau tampilan aplikasi.

2.16.3 Construction

Tahapan ini sistem yang akan dikembangkan dilakukan secara bertahap sesuai dengan rancangan desain dan model sistem. *Construction* dilakukan dengan

cara pengkodean aplikasi, pengujian awal aplikasi, dan perbaikan atau perubahan bila dibutuhkan.




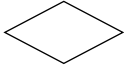


2.16.4 Cutover

Tahapan terakhir yakni sistem atau aplikasi yang sudah dibuat dapat digunakan oleh pengguna. Sistem atau aplikasi yang baru akan diterapkan setelah dilakukan pengujian secara keseluruhan dengan menggunakan metode pengujian.





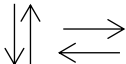
2.17 Flowchart

Flowchart merupakan simbol-simbol pekerjaan yang menunjukkan bagan aliran proses yang saling terhubung, setiap simbol *flowchart* melambangkan pekerjaan dan instruksinya. (Muhdar Abdurahman, Mudar Safi, 2019). Simbol-simbol *flowchart* dapat dilihat pada Tabel 1.

Tabel 1. Tabel Simbol *Flowchart*

Simbol	Nama	Keterangan
	<i>Terminator</i>	Simbol permulaan atau akhir dari suatu kegiatan
	<i>Processing</i>	Menyatakan suatu proses sistem yang dilakukan oleh komputer
	<i>Input/Output</i>	Menyatakan proses masukan dan keluaran tanpa tergantung dengan jenis peralatannya
	<i>Decision</i>	Menyatakan suatu kondisi yang menghasilkan dua kemungkinan.
	<i>Connector</i>	Digunakan untuk menghubungkan dari proses ke proses lainnya dalam halaman yang sama
	<i>Offline Connector</i>	Digunakan untuk menghubungkan dari proses ke proses lainnya dalam halaman yang berbeda

Tabel 1. (Lanjutan)


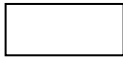
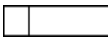
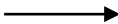
	<i>Predefined Process</i>	Digunakan untuk proses pelaksanaan suatu bagian (sub-program) atau prosedur
	<i>Punched Card</i>	Menyatakan <i>input</i> berasal dari kartu atau <i>output</i> berasal ditulis ke kartu
	<i>Punch Tape</i>	Menyatakan <i>input</i> atau <i>output</i> yang menggunakan pita kertas berlubang
	<i>Document</i>	Mencetak <i>output</i> dalam bentuk dokumen cetak
	<i>Flow Direction</i>	Menyatakan aliran arus suatu proses

Sumber: (Muhdar Abdurahman, Mudar Safi, 2019) (Rizky, 2019)

2.18 Data Flow Diagram (DFD)

Data Flow Diagram merupakan suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, dimana penggunaannya sangat membantu untuk memahami sistem logika, terstruktur dan jelas. (Dewi & Malfiany, 2017). Simbol-simbol *DFD* dapat dilihat pada Tabel 2.

Tabel 2. Tabel Simbol *Data Flow Diagram*

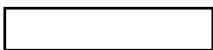


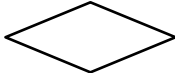



Simbol	Nama	Keterangan
	<i>Process</i>	Aktivitas yang dilakukan.
	<i>External entity</i>	Entitas yang berinteraksi dengan sistem, entitas berupa orang atau organisasi.
	<i>Data Storage</i>	Tempat penyimpanan data.
	<i>Data Flow</i>	Arah data yang akan mengalir

Sumber : (Dewi & Malfiany, 2017)

2.19 Entity Relationship Diagram (ERD)

Entity Relationship Diagram merupakan suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. *ERD* digunakan oleh professional sistem untuk berkomunikasi dengan pemakai eksekutif tingkat tinggi dalam suatu organisasi. (Saleha, 2016). Simbol-simbol *ERD* dapat dilihat pada Tabel 3.

Tabel 3. Tabel Simbol *Entity Relationship Diagram (ERD)*


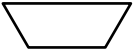

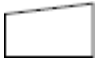
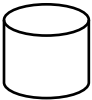



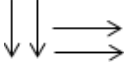


Simbol	Nama	Keterangan
	<i>Entity</i>	Objek riil yang dapat diidentifikasi oleh pengguna
	<i>Atribut</i>	Berfungsi untuk menerangkan entitas tersebut
	<i>Line</i>	Sebagai penghubung antara atribut dengan entitas maupun entitas dengan <i>relationship</i>
	<i>Relationship</i>	Berfungsi sebagai penghubung antara satu entitas atau lebih
	<i>Relasi 1:1</i>	Berfungsi untuk menunjukkan bahwa setiap entitas pada himpunan entitas pertama berhubungan dengan paling banyak satu entitas pada himpunan entitas kedua
	<i>Relasi 1:N</i>	Berfungsi untuk menunjukkan bahwa hubungan antara entitas pertama dengan entitas kedua adalah satu banding banyak atau sebaliknya. Setiap entitas dapat berelasi dengan banyak entitas pada himpunan entitas yang lain
	<i>Relasi N:N</i>	Berfungsi untuk menunjukkan bahwa setiap entitas pada himpunan entitas yang pertama dapat berhubungan dengan banyak entitas pada himpunan entitas yang kedua, demikian juga sebaliknya

Sumber : (Saleha, 2016)

2.20 Mapping chart

Mapping chart merupakan aliran data berbentuk dokumen atau formulir di dalam suatu sistem informasi yang merupakan suatu aktivitas yang saling terikat dalam hubungannya dengan kebutuhan data dan informasi. (Rahmatya & Faris, 2016). Simbol-simbol *Mapping Chart* dapat dilihat pada Tabel 4.

Tabel 4. Tabel simbol *Mapping Chart*

Simbol	Nama	Keterangan
	<i>Document</i>	Dokumen input dan output
	<i>Manual action</i>	Proses kegiatan sistem secara manual
	<i>Process</i>	Proses kegiatan sistem dari program komputer
	<i>Keyboard</i>	Input atau masukan yang terkomputerisasi
	<i>Hardisk storage</i>	Tempat penyimpanan data
	<i>Offline short symbol</i>	Proses dalam pengurutan data secara manual
	<i>Punched card symbol</i>	Proses input dengan kertas
	<i>Display</i>	Menampilkan data dilayar
	<i>Flow lines</i>	Aliran dari setiap proses
	<i>Connection</i>	Penghubung halaman satu ke halaman lain.
	<i>Communication link</i>	Menunjukkan hubungan dengan alat komunikasi

Sumber : (Rahmatya & Faris, 2016)

2.21 Blackbox Testing

Blackbox testing adalah pengujian yang dilakukan untuk mengetahui kesalahan fungsionalitas fitur pada sebuah aplikasi. Sebelum melakukan pengujian

blackbox yang harus diperhatikan adalah membuat daftar kebutuhan fungsional dan non-fungsional untuk mengetahui fitur-fitur apa saja yang akan diuji. Salah satu bentuk pengujian *blackbox* adalah pengujian validasi yang bertujuan untuk memeriksa apakah setiap fitur yang ada keluarannya atau *output* sesuai dengan yang diinginkan. (Affandi, 2019)

2.22 Artikel Terkait

Artikel terkait merupakan teori atau gagasan dari beberapa penelitian sebelumnya yang dapat menjadi acuan serta sebagai pendukung pada suatu penelitian. Adapun beberapa penelitian yang didapat penulis tentang aplikasi pengolahan data adalah sebagai berikut:

1. Putra et al., (2019), dalam jurnalnya yang berjudul “Sistem Informasi Dan Pengolahan Data Kriminal Di Polres Pasaman Barat Dengan Akses *Virtual Host* Berbasis *Client Server* Menggunakan Bahasa Pemrograman *PHP* Dan *Database Mysql*” dijelaskan bahwa proses pengolahan data pada Intelkam Polres Pasaman Barat masih dilakukan dengan *microsoft word* yang mengurangi efesiensi kerja dan mempersulit anggota kepolisian melakukan penanganan dalam pengolahan data kriminal. Hal ini mengalami kelemahan dalam pengolahan data yang tidak efektif dan efisien keterlambatan dalam pencarian data, bahkan data yang ada dapat mengalami kerusakan bahkan hilang. Berdasarkan permasalahan tersebut, maka penelitian ini merancang sistem informasi dan pengolahan data dengan menggunakan akses *Virtual Host* berbasis *Client Server* untuk dapat mengolah data kriminal dan mampu di implementasikan di tempat penelitian.

2. Siregar, H. F., Siregar, Y. H., & Melani, (2018), dalam jurnalnya yang berjudul “Perancangan Aplikasi Komik Hadist Berbasis Multimedia” dijelaskan bahwa pada masa sekarang ini kurangnya media yang memperlihatkan bagaimana berperilaku baik atau berakhlak yang baik dalam bergaul di tengah-tengah masyarakat berdasarkan hadist sangatlah minim, terutama untuk kalangan anak-anak maupun remaja. Berdasarkan permasalahan tersebut, maka penelitian ini membangun sebuah aplikasi komik hadist yang bertujuan untuk mengubah cara penyampaian dari aplikasi sebelumnya yang menyajikan komik secara manual menjadi komik semi-animasi dalam mendalami ilmu tentang agama.
3. Muslim & Dayana, (2016), dalam jurnalnya yang berjudul “Sistem Informasi Peraturan Daerah (Perda) Kota Pagar Alam Berbasis *Web*” dijelaskan bahwa pada saat itu sistem informasi tentang Peraturan Daerah Kota Pagar Alam belum berbasis *web*, sehingga dengan jumlah perda saat ini yang mencapai 166 perda, untuk memudahkan pencarian informasi bagi instansi terkait dan masyarakat luas tentang perda perlu dirancang suatu sistem informasi perda yang berbasis *web*. Berdasarkan permasalahan yang ada, maka peneliti mencoba untuk membuat sistem dengan menggunakan bahasa pemrograman *PHP* dan basis data *MySQL*, yang diharapkan bisa diimplementasikan pada sistem yang sebenarnya jika sistem ini dapat membantu pengaksesan data perda secara efektif, efisien dan keakuratan data dapat menjadi acuan pengambilan keputusan di waktu yang akan datang.
4. Ibrahim et al., (2018), dalam jurnalnya yang berjudul “Sistem Informasi Pengaduan Pelanggan Air Berbasis *Website* Pada PDAM Kota Ternate” dijelaskan bahwa Perusahaan Daerah Air Minum (PDAM) Kota Ternate

merupakan Badan Usaha Milik Daerah (BUMD) yang pelayanan informasi kepada pelanggan masih konvensional dimana pelanggan yang ingin mengadukan keluhannya harus datang ke kantor unit wilayah PDAM Kota Ternate. Proses pengaduan hanya dapat dilakukan pada hari kerja Senin - Jumat mulai dari Pukul 07.30 - 15.00 WIT. Diluar waktu tersebut pelanggan tidak dapat menyampaikan keluhannya sehingga dianggap kurang efektif. Berdasarkan permasalahan yang ada, maka peneliti membuat Sistem Informasi Pengaduan pelanggan air berbasis *Web* pada PDAM Kota Ternate agar mempermudah pelanggan dalam melakukan proses pengaduan air pada PDAM Kota Ternate agar meningkatkan kinerja dan pelayanan petugas kepada masyarakat yang ingin melakukan pengaduan air.

5. Novendri et al., (2019), dalam jurnalnya yang berjudul “Aplikasi Inventaris Barang Pada Mts Nurul Islam Dumai Menggunakan Php Dan Mysql” dijelaskan bahwa proses pencatatan inventaris barang pada MTS Nurul Islam Dumai masih dilakukan secara konvensional, itu menyebabkan informasi tentang inventaris sekolah sulit diketahui secara pasti dan cepat. Berdasarkan permasalahan yang ada, maka peneliti membuat sebuah aplikasi perancangan sistem informasi inventaris barang ini dengan tujuan untuk memudahkan pencatatan data barang yang cukup banyak dan rumit serta masih dikelola secara manual pada MTS Nurul Islam Kota Dumai agar dilakukan dengan lebih cepat dan akurat.
6. Sari, (2017), dalam jurnalnya yang berjudul “Rancang Bangun Aplikasi Penjualan Dan Persediaan Obat Pada Apotek Merben Di Kota Prabumulih” dijelaskan bahwa Apotek Merben masih menggunakan sistem manual dalam

penjualannya maupun pengelolaan data obatnya. Sistem manual itu adalah dengan menggunakan Alat Tulis Kantor (ATK) dalam penjualan dan pengelolaan data obatnya. Sehingga dibutuhkan sistem yang bisa membantu sistem penjualan dan pengelolaan data obatnya. Berdasarkan permasalahan yang ada, maka peneliti mengusulkan Aplikasi Penjualan dan Persediaan Obat pada Apotek Merben sehingga mampu menghasilkan informasi yang cepat, dan tepat waktu serta memudahkan pengolahan data penjualan obat dan data stok obatnya.

7. Lestanti & Susana, (2016), dalam jurnalnya yang berjudul “Sistem Pengarsipan Dokumen Guru Dan Pegawai Menggunakan Metode Mixture Modelling Berbasis *Web*” dijelaskan bahwa Pengelolaan data kearsipan pada SMK Negeri 1 Blitar masih menggunakan cara manual, dimana petugas masih mengarsipkan dokumen diorder. Berdasarkan permasalahan yang ada, maka peneliti menghasilkan aplikasi pengelolaan data kearsipan di SMK Negeri 1 Blitar dari sistem manual ke dalam sistem berbasis *web*, dengan hasil yang diharapkan dari penelitian adalah aplikasi pengelolaan data kearsipan yang dapat menghasilkan *output* seperti yang diharapkan dan proses pengarsipan dokumen dapat dilakukan dengan lebih baik, cepat, dan mudah.
8. Destiningrum & Adrian, (2017), dalam jurnalnya yang berjudul “Sistem Informasi Penjadwalan Dokter Berbasis *Web* Dengan Menggunakan *Framework Codeigniter* (Studi Kasus: Rumah Sakit Yukum Medical Centre)” dijelaskan bahwa sistem penjadwalan di Rumah Sakit Yukum Medical Centre masih menggunakan kertas apabila pasien ingin mengetahui informasi jadwal dokter dapat menanyakan bagian informasi. Berdasarkan permasalahan yang

ada, maka dari itu Sistem Informasi Penjadwalan Dokter Berbasis *Web* dapat menjadi solusi untuk pengolahan data serta dapat memberikan informasi tentang kesehatan dan jadwal dokter kepada pasien yang lebih akurat.

9. Fridayanthie & Charter, (2016), dalam jurnalnya yang berjudul “Rancang Bangun Sistem Informasi Simpan Pinjam Karyawan Menggunakan Metode *Object Oriented Programming* (Studi Kasus: PT Arta Buana Sakti Tangerang)” dijelaskan bahwa terdapat kendala yang telah berjalannya simpan pinjam karyawan ini diantaranya untuk melakukan pelaporan potongan simpanan wajib dan pembayaran angsuran pinjaman perlu melihat semua buku anggota dan menjumlahkan semua tagihan. Berdasarkan permasalahan yang ada, maka dibutuhkan sistem komputerisasi agar transaksi-transaksi dapat berjalan dengan baik dan cepat pada saat dibutuhkan. Penulis melakukan perancangan sistem informasi simpan pinjam karyawan dan mendesain tampilan dengan menggunakan *Integrated Development Environment (IDE) Netbeans* dengan bahasa pemrograman *Java* dan *MySQL* sebagai basis datanya.
10. Affandi, (2019), dalam jurnalnya yang berjudul “Rancang Bangun Smart Garden Berbasis Internet Of Thing (IoT) dengan Bot Telegram” dijelaskan bahwa Jurnal ini juga dalam upaya menyongsong *society 5.0* yang baru saja diperkenalkan di Jepang dengan cara yang *trending* di masa sekarang yaitu *Internet of Thing (IoT)*, semua dapat dikontrol dengan ponsel pintar yang terhubung dengan jaringan *internet*, ini dapat mempermudah manusia dalam melakukan berbagai hal seperti menyalakan peralatan elektronik di rumah, mengawasi atau memantau keamanan rumah, hingga sekarang mobil dapat di

kontrol menggunakan ponsel pintar. Ini membuktikan bahwa sekarang *internet* sudah menjadi bagian dari kehidupan dan akan terus berkembang.

11. Rahmatya & Faris, (2016), dalam jurnalnya yang berjudul “Pengembangan Sistem Informasi Akademik Berbasis *Web* Pada SMA Pasundan 2 Bandung” dijelaskan bahwa pengembangan untuk melaksanakan kegiatan mutasi masuk, mutasi keluar, penjurusan, penilaian yang masih bersifat manual dan belum terkomputerisasi dan penyimpanan datanya belum saling terintegrasi satu sama lain (basis data), sehingga dalam berbagai laporan kurang efektif dan efisien. Berdasarkan permasalahan yang ada, maka penelitian menunjukkan bahwa pengembangan sistem informasi akademik di SMA Pasundan 2 Bandung ini dapat membantu bagian kesiswaan, dalam mengolah data mutasi siswa keluar, mutasi siswa masuk, penjurusan dan penilaian. Mengadakan pengujian terhadap sistem aplikasi dengan menggunakan metode *blackbox*. Tahap akhir adalah mengimplementasikan sistem informasi akademik ini yang meliputi implementasi perangkat lunak, perangkat keras, dan basis data antarmuka dari aplikasi yang dihasilkan.