

BAB 2

LANDASAN TEORI

Bab ini memaparkan tentang penelitian-penelitian terdahulu yang berhubungan dengan topik Skripsi. Bab 2 ini juga menjelaskan teori-teori yang digunakan oleh penulis dalam pembuatan tugas akhir/Skripsi.

2.1 Penelitian Terdahulu

Dalam penelitian ini tidak lepas dari adanya penelitian terdahulu yang akan digunakan sebagai acuan oleh penulis untuk menyelesaikan permasalahan yang akan di teliti. Beberapa peneliti telah melakukan penelitian menggunakan objek masalah yang sejenis. Penelitian terdahulu yang dirujuk akan dijabarkan di bab ini.

Pertama yakni penelitian dengan judul “Notification System Chicken Feeder Automation Berbantuan Microcontroller Berbasis Php (Hypertext Preprocessor)” oleh (Ihsan, Ritzkal, & Adhi, 2018). Hasil dari penelitian Sistem dapat mengirimkan notifikasi ke pengguna berupa informasi – informasi mengenai jadwal, kuantiti pakan, kondisi tanki penyimpanan pakan dan kondisi wadah pakan melalui web dan whatsapp secara aktual.

Kedua yakni penelitian dengan judul “Monitoring Jadwal Kuliah Pada Iclass Melalui Notifikasi Bot Telegram” oleh (Safirman, 2019). Hasil dari penelitian Penggunaan aplikasi IM meningkat drastis dari tahun ke tahun, meskipun penggunaan IM masih belum dapat sepenuhnya menggantikan SMS, seiring dengan peningkatan pengguna perangkat seluler, permintaan data juga meningkat.

Ketiga yakni penelitian dengan judul “Aplikasi Scheduller Team Meeting Berbasis Mobile Dengan Menggunakan Push Notification” oleh (Masmur & Adven, 2021). Hasil dari penelitian aplikasi scheduler dapat digunakan

untuk melakukan pendwalan, pelacakan anggota yang melaksanakan pertemuan eksternal, digunakan oleh berbagai perangkat dan mempunyai tampilan yang responsif.

Keempat yakni penelitian dengan judul "Pengembangan Sistem Informasi Pencatatan Riwayat Dan Notifikasi Penjadwalan Donor Darah Berbasis Website Menggunakan Framework Codeigniter" oleh (Yani, Endang, & Uba, 2021). Hasil dari penelitian menunjukkan nilai rata-rata dari nilai Efficiency adalah 98,33%, nilai Usability 94,33%, nilai Correctness 95,56%, nilai Integrity 96,67%, dan nilai Testability 96,67%. dari hasil uji manfaat tersebut dapat disimpulkan bahwa pengembangan sistem ini dapat diterapkan dan digunakan untuk mempermudah poses pencatatan riwayat donor darah dan notifikasi penjadwalan donor darah di UDD PMI Kabupaten Banyumas.

Kelima yakni penelitian dengan judul "Rancang Bangun Aplikasi Mobile Penjadwal Perkuliahan Dengan Firebase Dengan Realtime Notification" oleh (Yulia & Muntahanah, 2021). Hasil dari penelitian menunjukkan Aplikasi yang dibuat menghadirkan Pemberitahuan secara otomatis sehingga dapat mempermudah mahasiswa untuk mendapatkan jadwal perkuliahan langsung dari dosen yang bersangkutan.

Keenam dari penelitian berjudul "Sistem Informasi Pendaftaran Vaksin Covid-19 Berbasis Website Memanfaatkan Whatsapp Gateway" oleh (Nurul Fitriani, 2022). Hasil dari penelitian menunjukkan bahwa Aplikasi yang dibuat bisa mengelola agenda vaksin dan mengirimkan pesan berupa jadwal agenda ke akun Whatsapp pendaftar vaksin secara otomatis melalui sistem dan secara broadcast.

Dari penelitian terdahulu yang di uraikan diatas masih terdapat kekurangan dimana pengiriman notifikasi masih kurang disesuaikan dengan target aplikasi yakni dengan memanfaatkan aplikasi yang banyak dan umum dipakai yakni Whatsapp untuk menyebar informasi agenda acara yang dibuat, oleh karena itu penulis membuat Sistem Otomasi



Monitoring Penjadwalan Agenda Berbasis Whatsapp Notification memanfaatkan *Whatsapp Gateway* yang dapat memberikan informasi isi agenda dengan lengkap dan cepat melalui *web* dan notifikasi pada *Whatsapp* yang sudah tercantum sebagai penerima notif. Berikut ini adalah tabel penelitian terdahulu sebagai rujukan penelitian Sistem Otomasi *Monitoring Penjadwalan Agenda Berbasis Whatsapp Notification* ini pada Tabel 2.1



Tabel 2.1 Penelitian Terdahulu

NO	Judul Penelitian	Penulis	Tahun Penelitian	Perbedaan Sistem	Persamaan Sistem	Output Sistem
1	Notification System Chicken Feeder Automation Berbantuan Microcontroller Berbasis Php (Hypertext Preprocessor)	Ihsan, Ritzkal, dan Bayu.	2018	Pada sistem ini tidak dijelaskan jangka waktu notif dikirimkan	Menggunakan <i>web</i> untuk mengolah dan menjadwalkan informasi yang dikirim melalui Whatsapp	Sistem yang mengirimkan notifikasi ke pengguna melalui web dan whatsapp secara aktual.
2	Monitoring Jadwal Kuliah Pada Iclass Melalui Notifikasi Bot Telegram	Emanuel Safirman Bata	2019	Sistem ini memanfaatkan fitur BOT pada telegram	Memfaatkan aplikasi <i>chatting</i> global.	Sistem yang membantu mahasiswa untuk mendapatkan informasi jadwal kuliah secara realtime melalui Bot Telegram
3	Aplikasi Scheduller Team Meeting Berbasis Mobile Dengan Menggunakan Push Notification	Masmur dan Adven	2021	Sistem ini melakukan pelacakan subjek untuk merangkai kegiatan rapat	Mengelola Jadwal Agenda	scheduller dapat melakukan pendwalan, pelacakan anggota yang melaksanakan pertemuan eksternal

Lanjutan Tabel 2.1

NO	Judul Penelitian	Penulis	Tahun Penelitian	Perbedaan Sistem	Persamaan Sistem	Output Sistem
4	Pengembangan Sistem Informasi Pencatatan Riwayat Dan Notifikasi Penjadwalan Donor Darah Berbasis Website Menggunakan Framework Codeigniter	Rosalina, Endang, dan Angela	2021	Sistem memberikan informasi penjadwalan melalui laman website yang dibuat	Memakai website untuk menjadwalkan kegiatan	pengembangan sistem digunakan untuk mempermudah poses pencatatan riwayat donor darah dan notifikasi penjadwalan donor darah
5	Rancang Bangun Aplikasi Mobile Penjadwal Perkuliahan Dengan Firebase Dengan Realtime Notification	Yulia dan Muntahanah	2021	Sistem yang dibuat dan berjalan pada perangkat yang ditargetkan	Membangun aplikasi yang dapat mengelola penjadwalan	Aplikasi menghadirkan Pemberitahuan secara otomatis sehingga dapat mempermudah mahasiswa untuk mendapatkan jadwal perkuliahan langsung dari dosen yang bersangkutan

Lanjutan tabel 2.1

NO	Judul Penelitian	Penulis	Tahun Penelitian	Perbedaan Sistem	Persamaan Sistem	Output Sistem
6	Sistem Informasi Pendaftaran Vaksin Covid-19 Berbasis Website Memanfaatkan Whatsapp Gateway	Nurul Fitriani	2022	Sistem bisa secara masal mengirimkan jadwal	Sistem bisa menampilkan jadwal dan mengirimkan agenda ke pengguna yang terdaftar	Sistem dapat mengirimkan notifikasi ke pengguna melalui web dan whatsapp secara aktual.

2.2 Kajian Pustaka

Bagian ini memuat rangkuman teori-teori yang diambil dari buku atau literatur yang mendukung penelitian, serta memuat penjelasan tentang konsep dan prinsip dasar yang diperlukan untuk pemecahan permasalahan. Landasan teori dapat berbentuk uraian kualitatif, model matematis, atau tools yang langsung berkaitan dengan permasalahan yang diteliti. Sumber teori yang dirujuk pada bagian ini harus dicantumkan dalam kalimat atau pernyataan yang diacu dan dicantumkan dalam Daftar Pustaka. Adapun dalam penulisan di dalam Landasan Teori disesuaikan dengan tema yang telah ditentukan.

2.2.1 Sistem Informasi

Sistem informasi terdiri dari beberapa komponen. Dalam buku “Prinsip-Prinsip Sistem Informasi Manajemen”, pengertian sistem informasi adalah sistem informasi yang diciptakan oleh para pengolahan data anak, data kegiatan, data *sponsor*, data pemeriksaan kesehatan, pemeriksaan *sosio* emosi, data perpindahan anak, data pejabat, data *tutor*, bantuan pelayanan dan data *inventarisasi* barang *analisis* dan manajer guna melaksanakan tugas khusus tertentu yang sangat *esensial* bagi berfungsinya organisasi.

Rancangan Sistem Informasi adalah merancang atau membuat sistem baru yang diterapkan untuk mengatasi masalah yang lama. Perancangan sistem dapat diartikan sebagai tahap setelah *analisis* dari *sirkulus* pengembangan sistem, pendefinisian dari kebutuhan *fungsionalis*, persiapan untuk rancangan bangunan implementasi, mengGambarkan bagaimana suatu sistem dibentuk (pengGambarkan, perencanaan, pembatasan sketsa) termasuk mengkonfigurasi komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem (Helpi, 2018).

2.2.2 Penjadwalan

Menurut (Yulia & Muntahanah, 2021) Penjadwalan merupakan pengaturan waktu dari suatu kegiatan operasi. Penjadwalan mencakup kegiatan mengalokasikan fasilitas, peralatan ataupun tenaga kerja bagi suatu kegiatan operasi dan menentukan urutan pelaksanaan kegiatan operasi. Dalam hierarki pengambilan keputusan, penjadwalan merupakan langkah terakhir sebelum dimulainya operasi. Tujuan penjadwalan untuk meminimalkan waktu proses, waktu tunggu langganan, dan tingkat persediaan, serta penggunaan yang efisien dari fasilitas, tenaga kerja, dan peralatan.

Kebanyakan perusahaan menyelesaikan pekerjaan secara bersamaan, karena itu perlu menggabungkan beberapa jadwal kerja. Penggabungan ini dimungkinkan apabila tanggal penyerahan atau selesai untuk setiap pekerjaan dapat diketahui dan seluruh penggabungan tersebut akan dilaksanakan oleh setiap bagian proses sepanjang periode yang direncanakan. Proses penggabungan ini disebut Penjadwalan (*scheduling*) dan hasilnya secara sederhana disebut jadwal (*schedule*) atau jadwal produksi (*production schedule*) secara keseluruhan. Salah satu kunci keberhasilan dalam meningkatkan efisiensi dalam unit operasi adalah kemampuan untuk menyusun jadwal secara efektif.

Beberapa tujuan yang ingin dicapai dengan dilaksanakannya penjadwalan adalah sebagai berikut :

- 1) Meningkatkan produktivitas mesin, yaitu dengan mengurangi waktu mesin menganggur.
- 2) Mengurangi persediaan barang setengah jadi dengan mengurangi jumlah rata-rata pekerjaan yang menunggu antrian suatu mesin karena mesin tersebut sibuk.
- 3) Mengurangi keterlambatan karena telah melampaui batas waktu
- 4) Meminimasi ongkos produksi.
- 5) Pemenuhan batas waktu yang telah ditetapkan (*due date*), karena dalam kenyataan apabila terjadi keterlambatan

pemenuhan due date dapat dikenakan suatu denda (*penalty*).

2.2.3 Whatsapp Gateway

Menurut (Ahmad, 2021) merupakan aplikasi pesan seluler lintas *platform* yang memungkinkan Anda untuk bertukar pesan tanpa harus membayar SMS. Pada awalnya aplikasi ini hanya menarik sebagian kecil orang. Namun setelah menambahkan fitur *messaging* aplikasi ini menjadi semakin populer

Whatsapp Gateway didefinisikan sebuah sistem aplikasi yang digunakan untuk mengirim pesan (*sending*) dan menerima pesan (*inbox*) *WhatsApp* dari *web* atau aplikasi lain ke perangkat *WhatsApp user*. Dengan *Whatsapp Gateway* setiap gawai dan aplikasi dapat terhubung dengan perangkat *Whatsapp* untuk saling bertukar informasi dan data.

Yang berperan dalam konektivitas tersebut adalah *Application Programming Interface* atau *API*. *API* berperan sebagai pembawa pesan yang menerima permintaan pengguna dan memberitahu sistem apa yang harus dilakukan, lalu memberikan *respons* yang sesuai untuk permintaan tersebut. Sehingga dengan *Whatsapp Gateway*, anda bisa melakukan integrasi ke sistem anda secara langsung dan melakukan pengiriman pesan *whatsapp personal* maupun pesan massal. Cara kerja *Whatsapp Gateway* adalah sebagai berikut :

- 1) Menerima pesan *Whatsapp* sesuai dengan *keyword* yang ditentukan.
- 2) Melakukan fungsi logik tertentu terhadap data-data yang diterima dari *Whatsapp Gateway*.
- 3) Mengirimkan informasi kepada *user* berdasarkan *keyword* sesuai dengan permintaan.



2.2.4 WEB

Web merupakan metode atau program yang dapat menampilkan beberapa informasi di *internet*, seperti informasi yang berupa teks, suara, Gambar maupun video yang *interaktif* dan memiliki kelebihan untuk menghubungkan (*link*) satu dokumen dengan dokumen lainnya (*hypertext*) dan dapat diakses melalui sebuah *browser*.

Dari pengertian-pengertian diatas dapat disimpulkan bahwa *Web* adalah suatu *aplikasi* berbasis internet yang terdiri dari kumpulan halaman yang saling terhubung dan digunakan untuk menampilkan informasi berupa teks, Gambar, suara, video dan program *multimedia* lainnya berupa animasi (tulisan bergerak, Gambar bergerak), dan atau gabungan dari semuanya (Supriady, Iwan, & Hardiansyah, 2018).

2.2.5 PHP

PHP (Hypertext Preprocessor) adalah bahasa pemrograman script yang paling banyak dipakai saat ini. *PHP* banyak dipakai untuk memrogram situs *web* dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain. Contoh terkenal dari aplikasi *PHP* adalah forum (*phpBB*) dan *MediaWiki* (*software* di belakang *Wikipedia*). *PHP* juga dapat dilihat sebagai pilihan *ASP.NET / C# / VB.NET Microsoft, ColdFusion Macromedia, JSP/Java Sun Microsystems, dan CGI/Perl*. Contoh aplikasi lain yang lebih kompleks berupa *CMS* yang dibangun menggunakan *PHP* adalah *Mambo, Joomla, Postnuke, Xaraya*, dan lain – lain (Aminulloh, Fibrian, & Masrur, 2020).

2.2.6 MySQL

MySQL(My Structured Query Language” adalah sebuah perangkat lunak sistem manajemen basis data *SQL*(bahasa Inggris: database management system) atau *DBMS*



yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. *RDBMS* adalah program yang memungkinkan pengguna *database* untuk membuat, mengelola, dan menggunakan data pada suatu model relational. Dengan demikian, Tabel-Tabel yang ada pada *database* memiliki relasi antara satu Tabel dengan Tabel lainnya.

MySQL adalah turunan konsep utama dalam *database* yaitu *SQL (Structured Query Language)*. *SQL* merupakan sebuah konsep untuk mengoperasikan *database*, terutama untuk seleksi atau pemilihan serta pemasukan data, yang memungkinkan proses operasional data dikerjakan secara otomatis dan mudah (Supriady, Iwan, & Hardiansyah, 2018).

2.2.7 UML

UML (Unified Modeling Language) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta mengGambarkan arsitektur dalam pemrograman berorientasi objek. *UML* muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, mengGambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. *UML* hanya berfungsi untuk melakukan pemodelan, jadi penggunaan *UML* tidak terbatas pada *metodologi* tertentu, meskipun pada kenyataannya *UML* paling banyak digunakan pada *metodologi* berorientasi objek. *UML* sendiri terdiri atas pengelompokkan diagram-diagram sistem menurut aspek atau sudut pandang tertentu

1. *Use Case* Merupakan Gambaran dari *fungsi* yang diharapkan dari sebuah sistem, dan merepresentasikan sebuah interaksi antara aktor dan sistem. Didalam *use case* terdapat *actor* yang merupakan sebuah Gambaran entitas dari manusia atau sebuah sistem yang melakukan pekerjaan di sistem.



Tabel 2.2 *Use Case*

Simbol	Keterangan
	Aktor : Seseorang atau sesuatu yang berinteraksi dengan sistem yang dikembangkan.
	<i>Use case</i> : perangkat teringgi dari fungsionalitas yang dimiliki sistem.
	<i>Association</i> : adalah relasi antara aktor dan <i>use case</i> .
	<i>Generalisasi</i> : untuk memperhatikan struktur pewaris yang terjadi.

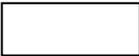
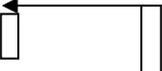
2. *Activity Diagram* Merupakan Gambaran alir dari aktivitas-aktivitas didalam sebuah sistem yang berjalan.

Tabel 2.3 *Activity Diagram*

Simbol	Nama
	Titik Awal
	Titik Akhir
	<i>Activity</i>
	Pilihan untuk pengambilan keputusan

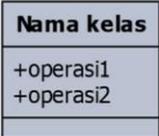
3. *Sequence Diagram* merupakan gambaran interaksi antar objek didalam dan di sekitar sistem yang berupa message yang digambarkan terhadap waktu.

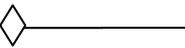
Tabel 2.4 *Sequence Diagram*

Simbol	Keterangan
	Aktor, mengGambarkan pengguna sistem
	<i>Lifeline</i> , objek antarmuka yang saling berinteraksi.
	<i>Message</i> , spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktifitas yang terjadi.
	<i>Message</i> , spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktifitas yang terjadi.
	<i>SelfMessage</i> , mengGambarkan aktifitas pesan pada aktor itu sendiri.

4. *Class Diagram* : Merupakan Gambaran struktur dan deskripsi dari *class*, *package*, dan objek yang saling berhubungan seperti diantaranya pewarisan, asosiasi dan lainnya.

Tabel 2.5 *Class Diagram*

Simbol	Notasi	Keterangan
	<i>Class</i>	<i>Class</i> diGambarkan sebagai kotak dengan 3 bagian berisi nama, <i>atribut</i> , <i>method</i> .

	<i>Association</i>	Asosiasi merupakan sebuah <i>relationship</i> paling umum antara 2 class.
	<i>Composition</i>	Jika class tidak berdiri sendiri, maka class memiliki relasi terhadap class tempat dia bergantung.
	<i>Dependency</i>	Kadangkala sebuah class menggunakan class yang lain. Hal ini disebut <i>dependency</i> .
	<i>Aggregation</i>	<i>Aggregation</i> mengindikasikan keseluruhan bagian <i>relationship</i> , disebut sebagai relasi "mempunyai sebuah" atau "bagian dari".
	<i>Generalization</i>	Sebuah relasi <i>Generalization</i> pada sebuah relasi <i>inheritance</i> pada konsep berorientasi objek.

2.2.8 Waterfall

Menurut (Putu, Widiarty, Viktor, & Sari, 2022) Model *Waterfall* yaitu metode pengembangan perangkat lunak yang bersifat sekuensial serta terdiri dari beberapa tahap yang saling berhubungan :

- 1) *Requirements Analysis and Definition* (Analisis dan Definisi Kebutuhan).

Pada tahap ini dilakukan pengumpulan informasi kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh aplikasi yang akan di bangun. Tahap ini akan dilakukan analisis pengguna menggunakan proses, analisis teknologi yang digunakan dan analisis informasi yang didapat.

2) *System and Software Design* (Perancangan Sistem dan Perangkat Lunak)

Pada tahap ini berguna untuk membuat desain interface aplikasi yang akan dibuat, rancangan akan disesuaikan dengan kebutuhan pengguna.

3) *Integration and System Testing* (Integrasi dan Pengujian Sistem)

Unit program/program individual diintegrasikan menjadi sebuah kesatuan sistem dan kemudian dilakukan pengujian. Dengan kata lain, pengujian ini ditujukan untuk menguji keterhubungan dari tiap-tiap fungsi perangkat lunak untuk menjamin bahwa persyaratan sistem telah terpenuhi.

4) *Operation and Maintenance* (Operasi dan Pemeliharaan)

Mengoperasikan program dilingkungannya sesuai dengan kebutuhan user dan melakukan *maintenance*. Biasanya fase ini merupakan siklus yang paling lama. Sistem diinstal dan dipakai. Pemeliharaan mencakup koreksi dan berbagai error yang tidak ditemukan pada tahap-tahap sebelumnya, perbaikan atas implementasi unit sistem dan pengembangan pelayanan sistem.

2.2.9 Black Box Testing

Menurut (Dini, Yunita, & Lusi, 2021) Ujicoba dilakukan untuk menguji keakuratan sistem antara *input* dan *output* pada sistem yang dibangun. Teknik yang digunakan dalam pengujian



remainder imunisasi anak yaitu teknik pengujian *Black Box*. Teknik pengujian ini dilakukan untuk mengetahui setiap fungsi pada setiap halaman sistem berfungsi atau tidak, dan *output* sesuai dengan *rule* yang ditentukan. Beberapa teknik testing yang tergolong dalam tipe *Black Box Testing* antara lain :

- 1) *Equivalence Partitioning* pada teknik ini, tiap inputan data dikelompokkan ke dalam grup tertentu, yang kemudian dibandingkan *output*-nya.
- 2) *Boundary Value Analysis* pada teknik *Boundary Value Analysis*, dilakukan *input*-an yang melebihi batasan sebuah data, contohnya untuk sebuah *input*-an harga barang, maka dapat dilakukan *testing* dengan menggunakan angka negatif yang tidak diperbolehkan dalam sebuah harga. Jika perangkat lunak berhasil mengatasi *input*-an yang salah tersebut, maka dapat dikatakan teknik ini telah selesai dilakukan.
- 3) *Cause Effect Graph* dalam teknik ini, dilakukan proses *testing* yang menghubungkan sebab dari sebuah *input*-an dan akibatnya pada *output* yang dihasilkan.

