

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1. Tinjauan Pustaka

Penelitian sebelumnya dilakukan di Stikes Husada Mandiri Kupang pada tahun 2018 dengan judul “Perancangan Sistem Informasi Repository Karya Tulis Ilmiah Pada Stikes Husada Mandiri Kupang”. Melakukan pengembangan dari sistem sebelumnya yang mana masih belum secara keseluruhan mengelola karya ilmiah mahasiswa secara digital serta meminimalisir alokasi tempat yang dibutuhkan untuk menyimpan berkas karya ilmiah mahasiswa yang kian bertambah. Tujuan lain penelitian ini yaitu untuk memberi kemudahan untuk mencari dan membaca referensi karya tulis ilmiah dari alumni sehingga dapat dijadikan referensi untuk penelitian dan meminimalisir kemungkinan pengajuan karya tulis ilmiah dengan judul yang sama. Dalam mengembangkan sistem ini, peneliti mengkombinasikan antara model proses *waterfall* dengan pemodelan analisis berorientasi objek dan kemudian diimplementasikan dengan bahasa pemrograman PHP dengan *framework* Codeigniter dan menggunakan MySQL sebagai *database*. Dampak jangka panjang dari pengarsipan digital dan bank data karya tulis ilmiah di STIKes Citra Husada Mandiri Kupang adalah memberikan kemudahan bagi para mahasiswa dalam mencari dan membaca referensi karya tulis dari para alumni. Hal ini memungkinkan mereka untuk menggunakan referensi tersebut sebagai acuan. Selain itu, penggunaan sistem tersebut juga dapat meminimalkan kemungkinan pengajuan karya tulis dengan judul yang sama, karena dosen pembimbing dapat mengakses sistem untuk memeriksa ketersediaan judul tersebut [4].

Penelitian sebelumnya juga dilakukan di STMIK Syaikh Zainuddin NW Anjani pada tahun 2020 dengan judul “Sistem Informasi Repositori Skripsi Berbasis Web Pada STMIK Syaikh Zainudin NW Anjani”. Proses pengumpulan dan publikasi mahasiswa seperti skripsi masih dilakukan dengan cara yang konvensional, yaitu mahasiswa harus menyerahkan *file* skripsi menggunakan CD. Keterbatasan jumlah dan akses ke arsip perpustakaan menjadi kendala selanjutnya yang membuat mahasiswa harus menunggu dan mengantri. Penelitian ini bertujuan untuk membuat sebuah sistem yang mengubah proses pengumpulan file skripsi yang berbentuk softcopy tidak lagi dilakukan secara

konvensional, melainkan dikumpulkan melalui sistem agar nantinya skripsi tersebut bisa dicari kembali oleh mahasiswa sebagai referensi suatu penelitian dikemudian hari. Sistem ini diimplementasikan dengan bahasa pemrograman PHP dengan *framework* Codeigniter. Kemudian, dikembangkan menggunakan model proses pengembangan perangkat lunak *waterfall*. Fokus pada desain sistem yang akan dibangun meliputi rancangan UML yang terdiri dari *Use Case Diagram*, *Activity Diagram*, rancangan basis data, dan rancangan antarmuka aplikasi. Pengumpulan berkas skripsi dalam bentuk *softcopy* tidak lagi mengikuti metode konvensional. Saat ini, skripsi dapat diunggah melalui sistem, sehingga pencarian skripsi menjadi lebih mudah dan dapat diunduh secara *online*. Pemustaka yang ingin mencari skripsi lama sebagai referensi penelitian tidak perlu lagi pergi ke perpustakaan dan mencari di rak arsip [5].

Penelitian serupa dilakukan di SMK Citra Bangsa Bogor pada tahun 2022 dengan judul “Perancangan Sistem Pengarsipan Dokumen Administrasi Perangkat Pembelajaran Guru Berbasis Web di SMK Citra Bangsa Bogor”. Proses pengarsipan sebelum dilakukan penelitian ini, pengumpulan dokumen *softfile* harus diunggah melalui *link* Google Form, kemudian dokumen *hardfile* disimpan pada *box file* yang berada di lemari tata usaha. Karena guru-guru memiliki banyak dokumen administrasi perangkat pembelajaran yang perlu dikelola, maka perlu sebuah pengelolaan arsip yang baik. Dari masalah tersebut, penelitian ini bertujuan untuk mempermudah guru dalam mengelola dokumen terutama dalam mengatur penyimpanan dokumen dan dapat saling bertukar informasi mengenai dokumen administrasi perangkat pembelajaran guru secara cepat. Penelitian ini dirancang dengan berbasis *website* dengan metode *waterfall* dan menggunakan bahasa pemrograman PHP serta MySQL sebagai basis datanya. Sistem ini mempermudah guru dalam mengelola dokumen terutama dalam mengatur penyimpanan dokumen, dan dapat saling bertukar informasi mengenai dokumen administrasi perangkat pembelajaran guru. Dengan adanya sistem ini, guru-guru dimudahkan untuk mencari dokumen yang dibutuhkan dan dapat melihat dokumen antar guru yang dibutuhkan [6].

Pada penelitian ini, akan dibangun sebuah sistem informasi repositori karya tulis ilmiah untuk mempermudah pengguna dalam mengelola dan juga memudahkan pengaksesan karya tulis ilmiah yang nantinya bisa digunakan sebagai acuan sebuah penelitian. Pada sistem yang akan dibangun saat ini nantinya selain yang terkait dengan pengelolaan karya tulis ilmiah yaitu pengguna dapat melihat kartu

anggota perpustakaan secara digital, pengguna dapat mencari halaman pengesahan untuk pengajuan berkas jabatan fungsionalnya, pengguna dapat mencetak surat keterangan penyerahan laporan, dan pengguna dapat mengetahui pemustaka baik yang sudah atau belum mengumpulkan karya tulis ilmiah pada masing-masing program studi. Adapun pada sistem ini dibangun dengan menggunakan bahasa pemrograman *Golang* dan *React*. Sistem ini juga dilengkapi dengan fitur notifikasi *email*.

2.2. Landasan Teori

2.2.1. Karya Tulis Ilmiah

Karya tulis ilmiah adalah sebuah tulisan yang berisi tentang hasil penelitian, eksplorasi ide, atau pembahasan isu-isu tertentu secara sistematis dan berdasarkan metode ilmiah yang objektif. Karya tulis ilmiah umumnya ditulis dengan menggunakan bahasa yang formal dan memiliki struktur penulisan yang jelas, terdiri dari bagian pendahuluan, tinjauan literatur, metodologi, hasil penelitian atau pembahasan, dan kesimpulan [7].

2.2.2. Repositori

Repositori adalah suatu tempat penyimpanan digital untuk dokumen, data, dan kode sumber. Repositori ini dapat digunakan oleh berbagai jenis organisasi atau komunitas untuk mengatur, menyimpan, dan berbagi informasi. Jenis-jenis repositori digital antara lain adalah repositori terbuka (*open repository*), institutional repositori, dan data repositori [8].

2.2.3. Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan [9]. Berikut merupakan keomponen-komponen yang ada pada sistem informasi :

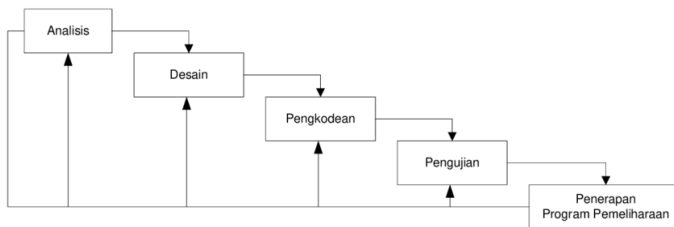
a. Blok masukan (*input block*)

Input mewakili data yang masuk kedalam sistem informasi. *Input* yang dimaksud adalah metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

- b. Blok model (*model block*)
Blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.
- c. Blok keluaran (*output block*)
Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem
- d. Blok teknologi (*technology block*)
Teknologi merupakan “*tool box*” dalam sistem informasi. teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu manusia (*brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).
- e. Blok basis data (*control block*)
Basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu sama lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*).
- f. Blok kendali (*control block*)
Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, ketidakefisienan, sabotase, dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan tersebut dapat langsung cepat diatasi.

2.2.4. Model *Waterfall*

Model *waterfall* adalah proses pengembangan perangkat lunak sekuensial di mana prosesnya seperti mengalir semakin ke bawah (mirip air terjun) melalui daftar tahapan yang harus dijalankan untuk membangun perangkat lunak komputer. Model *waterfall* mendefinisikan beberapa tahapan urutan yang harus diselesaikan satu demi satu dan kemudian pindah ke tahap berikutnya hanya ketika tahapan sebelumnya benar-benar selesai [2]. Diagram *waterfall* dapat dilihat pada gambar 2.1.



Gambar 2.1 Diagram *Waterfall*

Proses dari model *waterfall* antara lain:

1. Analisis Kebutuhan
Proses intensif dilakukan untuk mengumpulkan kebutuhan guna memperinci jenis kebutuhan perangkat lunak yang diinginkan oleh pengguna. Pada tahap ini, spesifikasi kebutuhan perangkat lunak harus didokumentasikan. Tujuannya adalah untuk memudahkan pemahaman mengenai perangkat lunak yang diperlukan oleh pengguna.
2. Desain
Desain program perangkat lunak melibatkan serangkaian proses berjenjang yang berfokus pada merancang program perangkat lunak, termasuk struktur data, arsitektur perangkat lunak, antarmuka pengguna, dan prosedur pengkodean. Langkah ini bertujuan untuk menerjemahkan kebutuhan perangkat lunak dari tahap analisis kebutuhan menjadi representasi desain agar dapat diimplementasikan pada tahap berikutnya.
3. Pengkodean
Melakukan implementasi dari tahap pembuatan desain dengan melakukan penulisan kode program dari desain yang sudah direncanakan.

4. **Pengujian**
Pengujian perangkat lunak bertujuan untuk mengevaluasi secara logis dan fungsional setiap komponennya, sehingga memastikan seluruh bagian yang diuji berfungsi dengan baik. Tujuan dari pengujian ini adalah untuk meminimalkan kesalahan dan memastikan bahwa hasil yang dihasilkan sesuai dengan yang diinginkan.
5. **Penerapan Program Pemeliharaan**
Ada kemungkinan bahwa perangkat lunak akan mengalami perubahan yang mana tidak sesuai dengan seharusnya setelah dikirim ke pengguna. Perubahan ini dapat terjadi karena kesalahan yang tidak terdeteksi selama pengujian atau perangkat lunak perlu menyesuaikan diri dengan lingkungan yang berbeda. Tahap dukungan atau pemeliharaan dapat memulai kembali proses pengembangan dari analisis spesifikasi untuk membuat perubahan pada perangkat lunak yang sudah ada, tetapi tidak digunakan untuk membuat perangkat lunak baru.

2.2.5. Basis Data

Basis data merupakan sebuah himpunan data yang terhubung dan disimpan bersama di dalam suatu media, tanpa adanya struktur data yang mengikat sehingga memudahkan penggunaannya kembali. Basis data dapat dioptimalkan oleh satu atau beberapa program aplikasi, dan data yang tersimpan tidak bergantung pada program yang akan menggunakannya. Selain itu, data dalam basis data disimpan dengan cara yang memungkinkan penambahan, pengambilan, dan modifikasi data dengan mudah dan terkontrol [10].

2.2.6. SQL (*Structured Query Language*)

SQL merupakan bahasa *query* standar yang digunakan untuk mengakses basis data relasional. Pernyataan SQL dapat dikelompokkan menjadi beberapa kelompok, yaitu DDL, DML, pengendali transaksi [11].

- a. **DDL (*Data Definition Language*)**
DDL merupakan kelompok perintah yang berfungsi untuk mendefinisikan atribut-atribut basis data, tabel, atribut (kolom), batasan-batasan terhadap suatu atribut, serta hubungan antar tabel. Yang termasuk dalam kelompok DDL ini adalah CREATE, ALTER, dan DROP.

- b. DML (*Data Manipulation Language*)
 DML adalah kelompok perintah yang berfungsi untuk memanipulasi data dalam basis data, misalnya untuk pengambilan, penyisipan, perubahan, dan penghapusan data. Perintah yang termasuk kategori DML adalah:
- SELECT : Memilih data
 - INSERT : Menambah data
 - DELETE : Menghapus data
 - UPDATE : Mengubah data
- c. Pengendali Transaksi
 Pengendali transaksi adalah perintah-perintah yang berfungsi untuk mengedalikan pengekseskusion transaksi. Perintah yang termasuk dalam kategori ini adalah:
- COMMIT : Menyetujui rangkaian perintah yang berhubungan erat (disebut transaksi) yang telah berhasil dilakukan.
 - ROLLBACK : Membatalkan transaksi yang dilakukan karena adanya kesalahan atau kegagalan pada salah satu rangkaian perintah.

2.2.7. MySQL

MySQL adalah sebuah sistem manajemen basis data relasional (RDBMS) yang berbasis *open-source*. MySQL digunakan untuk menyimpan, mengelola, dan mengakses data dari berbagai aplikasi, baik untuk keperluan web maupun desktop. MySQL sangat populer digunakan sebagai basis data dalam pengembangan aplikasi web, khususnya pada *platform* LAMP (Linux, Apache, MySQL, dan PHP) [12].

2.2.8. Golang

Golang merupakan bahasa pemrograman *opensource* yang dibuat di Google oleh Robert Griesemer, Rob Pike, dan Ken Thompson. Google merilis bahasa golang versi 1.0 pada Maret tahun 2012. Bahasa golang ditulis dalam *plain text*, kemudian dikompilasi menjadi bahasa mesin oleh *go compiler* untuk menghasilkan sebuah *executable binary* [13].




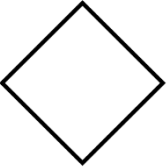
2.2.9. REST API


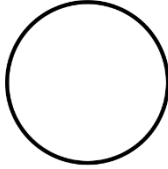
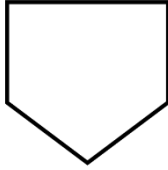
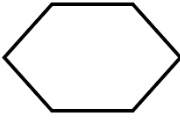
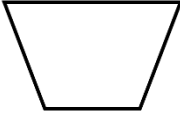

REST merupakan sebuah *stateless*, *cacheable*, dan arsitektur sederhana yang bukan merupakan sebuah protokol, melainkan sebuah *pattern* atau pola. REST API mengizinkan berbagai sistem untuk saling berkomunikasi mengirim atau menerima data dengan cara yang sederhana. Masing-masing pemanggilan REST API memiliki hubungan antara sebuah kata kerja HTTP dan URL [14].

2.2.10. Flowchart

Bagan alir atau *flowchart* merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur dalam sebuah program. Simbol-simbol yang digunakan dalam *flowchart* berguna untuk mempermudah dalam mendefinisikan proses. Proses dalam *flowchart* digambarkan secara vertikal dari atas ke bawah atau horizontal dari kiri ke kanan. Setiap aktivitas yang terdapat dalam gambaran tersebut harus didefinisikan dengan jelas agar dapat dipahami oleh pembaca [15]. Simbol-simbol *flowchart* dapat dilihat pada tabel 2.1.

Tabel 2.1 Simbol *Flowchart*

No	Gambar Simbol	Nama	Keterangan
1		Terminal	Simbol yang digunakan untuk menyatakan awal atau akhir suatu program
2		Arah aliran	Simbol yang digunakan untuk menunjukkan arah aliran proses
3		Proses	Simbol yang digunakan untuk menggambarkan pengolahan data
4		Keputusan	Simbol yang digunakan untuk menyatakan suatu pilihan berdasarkan suatu kondisi tertentu

5		Input/Output	Simbol yang digunakan untuk menunjukkan operasi masukan atau keluaran
6		Penghubung ke halaman yang sama	Simbol yang digunakan untuk menghubungkan bagian diagram alir pada halaman yang sama
7		Penghubung ke halaman lain	Simbol yang digunakan untuk menghubungkan bagian diagram alir pada halaman yang berbeda
8		Perisapan atau Preparation	Simbol yang digunakan untuk memberikan nilai awal pada suatu variabel atau pencacah
9		Operasi manual	Simbol yang menunjukkan pekerjaan yang dilakukan secara manual
10		Dokumen	Simbol yang menunjukkan dokumen yang digunakan untuk input dan output baik secara manual, mekanik, maupun komputerisasi

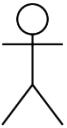
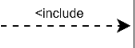
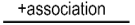
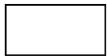

2.2.11. UML (*Unified Modeling Language*)

Unified Modeling Language merupakan sebuah standar penulisan atau blueprint yang mencakup bisnis proses dan penulisan kelas-kelas dalam bahasa yang spesifik. Tujuan utama dari UML adalah memberikan panduan dalam mengembangkan perangkat lunak yang terstruktur dan dapat diimplementasikan secara efisien [15].

A. *Use Case Diagram*

Penggunaan *Use Case* adalah untuk menggambarkan fungsi dasar dari suatu sistem informasi. *Use Case* memberikan deskripsi mengenai interaksi sistem bisnis dengan lingkungannya [15]. Simbol-simbol *Use Case* dapat dilihat pada tabel 2.2.


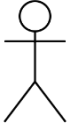
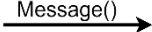
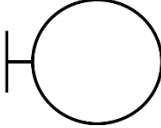
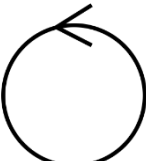
Tabel 2.2 Simbol *Use Case Diagram*

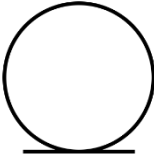
No	Simbol	Fungsi
1		Entitas luar sistem yang berinteraksi dengan sistem. Aktor biasanya merepresentasikan pengguna, perangkat, atau sistem lain yang terlibat dalam penggunaan sistem.
2		Keterkaitan antara dua atau lebih use case dimana satu use case menggunakan fungsi dari use case lain.
3		Koneksi atau hubungan antara aktor dan use case yang menunjukkan bahwa aktor terlibat dalam use case.
4		Batasan sistem yang dimana menunjukkan lingkup sistem dalam use case diagram.
5		Deskripsi interaksi antara pengguna atau aktor dan sistem. Use case menggambarkan aktivitas atau fungsi yang dilakukan oleh sistem dalam skenario tertentu.

B. *Sequence Diagram*

Sequence diagram menunjukkan pesan yang lewat di antara objek untuk *use case* tertentu dari waktu ke waktu. Sequence diagram mengilustrasikan objek-objek yang berpartisipasi di dalam suatu *use case* [15]. Simbol-simbol *Sequence* dapat dilihat pada tabel 2.3.

Tabel 2.3 Simbol *Sequence*

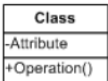
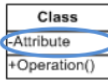
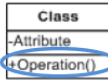
No	Simbol	Fungsi
1		Mewakili proses durasi aktivasi sebuah operasi
2		Entitas yang terlibat dalam skenario dan berinteraksi dengan sistem
3		Digunakan untuk berkomunikasi antar objek yang menggambarkan aksi yang akan dilakukan.
4		Menangani komunikasi antar lingkungan sistem
5		Bertanggung jawab terhadap kelas-kelas terhadap objek yang berisi logika

6		Gambaran sistem sebagai landasan dalam menyusun basis data
---	---	--

C. *Class Diagram*

Class diagram adalah model statis yang menunjukkan kelas dan hubungan di antara kelas yang tetap konstan dalam sistem dari waktu ke waktu. Class diagram menggambarkan kelas, yang meliputi perilaku dan keadaan, dengan hubungan antar kelas [15]. Simbol-simbol *Class Diagram* dapat dilihat pada tabel 2.4.

Tabel 2.4 Simbol *Class Diagram*

No	Simbol	Fungsi
1		Class, merepresentasikan orang, tempat, atau objek lain yang diperlukan sistem untuk menyimpan informasi. Nama kelas biasanya ditulis dengan huruf tebal dan terletak di bagian atas kotak. Atribut kelas terletak di kotak tengah dan operasi di kotak bawah. Meskipun demikian, kelas tidak secara spesifik menunjukkan operasi yang tersedia untuk semua kelas secara eksplisit.
2		Atribut, merupakan properti yang menggambarkan keadaan suatu objek. Dapat diturunkan dari atribut lain, ditampilkan dengan menempatkan garis miring sebelum nama atribut.
3		Operation, representasi dari tindakan atau fungsi yang dapat dijalankan oleh sebuah kelas dalam pemrograman. Dapat dikelompokkan menjadi tiga jenis, yaitu konstruktor, permintaan, atau operasi pembaruan.