

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Banyak penelitian yang membahas suatu pengembangan *website* menggunakan framework, diantaranya menggunakan framework ExpressJs, Laravel maupun Codengiter. Dari penelitian yang sudah dilakukan, penggunaan framework dalam pengembangan suatu *website* memiliki keunggulan dalam melakukan proses *development*, seperti efisien waktu dan mempermudah dalam proses *development*. Dari setiap framework memiliki kelebihan dan kekurangan sehingga pemilihan suatu framework disesuaikan dengan kebutuhan aplikasi yang akan dibangun. Berikut beberapa penelitian yang memiliki keterkaitan dengan penelitian yang akan dilakukan oleh penulis.

Pertama, penelitian dengan judul “Pengujian Portal Mahasiswa pada Sistem Informasi Akademik (SINA) menggunakan Black Box Testing dengan metode Equivalence Partitioning dan Boundray Value Analysis”. Penelitian dilakukan oleh Taufik Hidayat dan Hendar Dini Putri pada tahun 2019. Penelitian tersebut dilakukan untuk memastikan perangkat lunak dapat berjalan sesuai dengan fungsionalitas yang diharapkan yaitu menggunakan Black Box Testing dengan metode Equivalence Partitioning dan Boundray Value Analysis.

Kedua, penelitian dengan judul “Implementasi *Scrum* Dalam Pengembangan sistem Informasi Jasa Desain Grafis” yang dilakukan oleh lalu Mutawali, Buyung Kurnia Fathoni, Hasyim Asyari pada tahun 2020. Penelitian dilakukan untuk menerapkan metode *scrum* sebagai kerangka kerja pada pengembangan sistem jasa desain grafis untuk mempermudah layanan jasa design dan pembuatan kemasan lebih mudah yang dilakukan oleh UMKM dan industry kreatif di Lombok Tengah. Hasil dari penelitian yaitu sistem informasi layanan jasa desain grafis, yang dapat berfungsi untuk melakukan pemesanan jasa desain.

Ketiga, penelitian dengan judul “Rancang Bangun *Rest API* Aplikasi Weshare Sebagai Upaya Mempermudah Pelayanan Donasi Kemanusiaan”. Penelitian tersebut dilakukan oleh Hasanuddin, Hari Asgar, dan Budi Hartono pada

tahun 2022. Penelitian dilakukan untuk merancang *Rest API* pada aplikasi Weshare yaitu pelayanan donasi online menggunakan NodeJs, ORM Sequelize dan Json Web Token (JWT) sebagai teknologi yang digunakan.

Keempat, penelitian dengan judul “Pengujian White-Box pada aplikasi *Debt Manager* berbasis android” oleh Demi Mardian, Neneng, Ajeng Savitri Puspaningrum, Alfiansyah Hasibuan, Medi Hermanto Tinambunan pada tahun 2022. Penelitian ini dilakukan untuk menguji aplikasi *Debt Manager* menggunakan metode *whitebox*. Hasil pengujian *whitebox* pada aplikasi *Debt Manager* telah dinyatakan lolos tanpa adanya *error*.

Kelima, penelitian dengan judul “Analisis Perbandingan Performa Web Service Rest Menggunakan Framework Laravel, Django dan Node Js Pada Aplikasi Berbasis *Website*” oleh Abi Amarulloh, kurniasih dan Muchlis pada tahun 2023. Penelitian ini dilakukan untuk pengujian kecepatan *request* data per detik dan kemudian akan menampilkan performa web service yang tinggi pada aplikasi client berbasis *website*.

Tabel 2.1 Ringkasan Penelitian Sebelumnya

No	Judul	Penulis	Tahun	Studi kasus	Metode / Algoritma	Hasil	Perbedaan dengan penelitian yang dilakukan
1	Pengujian Portal Mahasiswa pada Sistem Informasi Akademik (SINA) menggunakan Black Box Testing dengan metode Equivalence Partitioning dan Boundray Value Analysis	Demi Mardian, Neneng, Ajeng Savitri Puspaningrum, Alfiansyah Hasibuan, Medi Hermanto Tinambunan	2019	Sistem Informasi Akademik Universitas Islam Syekh Yusuf (SINA)	Menggunakan Black Box Testing dengan metode Equivalence Partitioning dan Boundray Value Analysis.	Jawaban dari mahasiswa dapat diambil sebuah keputusan untuk pengembangan sistem informasi akademik Universitas Islam Syekh Yusuf (SINA). Pihak <i>development</i> dapat mempersiapkan apa yang masih harus dikembangkan.	Penelitian sebelumnya menggunakan <i>balckbox</i> testing untuk mengambil keputusan dengan metode Equivalence Partitioning dan Boundray Value Analysis untuk sistem informasi akademik, Sedangkan untuk penelitian yang akan dilakukan penulis yaitu untuk mengambil keputusan dengan blackboctx testing dengan

No	Judul	Penulis	Tahun	Studi kasus	Metode / Algoritma	Hasil	Perbedaan dengan penelitian yang dilakukan
							metode Equivalence Partitioning .
2	Implementasi <i>Scrum</i> Dalam Pengembangan sistem Informasi Jasa Desain Grafis	Ialu Mutawali, Buyung Kurnia Fathoni, Hasyim Asyari	2020	Sistem Informasi Jasa Desain Grafis di Lombok Tengah.	Menggunakan metode Scrum.	Sistem Infromasi layanan jasa design yang ada di Lombok Tengah dapat digunakan untuk melakukan pemesanan jasa desain.	Penelitan sebelumnya memiliki tujuan untuk Sistem Infromasi layanan jasa design yang ada di Lombok Tengah , Sedangkan penelitian yang akan dilakukan penulis akan diterapkan pada <i>Rest API</i> untuk aplikasi Desa Bersinar.
3	Rancang Bangun <i>Rest API</i> Aplikasi Weshare Sebagai Upaya	Hasanuddin, Hari Asgar, dan Budi Hartono	2022	Bangun <i>Rest API</i> Aplikasi Weshare	Menggunakan NodeJs, ORM Sequelize dan	Menghasilkan <i>Rest API</i> untuk di implementasikan pada client <i>website</i>	Penelitian sebelumnya yaitu pengembangan <i>Rest API</i> untuk transaksi pelayanan donasi,

No	Judul	Penulis	Tahun	Studi kasus	Metode / Algoritma	Hasil	Perbedaan dengan penelitian yang dilakukan
	Mempermudah Pelayanan Donasi Kemanusiaan				Json Web Token (JWT)	dan <i>mobile</i> WeShare terkait transaksi pelayanan donasi.	sedangkan penelitian yang akan dilakukan oleh penulis untuk mengembangkan <i>Rest API</i> pada Aplikasi Desa Bersinar.
4	Pengujian White-Box pada aplikasi <i>Debt Manager</i> berbasis android	Harya Gusdevi, Sri Kuswayati, Mohammad Iqbal Abu Bakar, Nesha Novianti, Rizky Ramadan	2022	Impelementasi <i>whitebox</i> pada aplikasi <i>debt manager</i>	Menggunakan metode <i>whitebox testing</i>	Hasil pengujian <i>whitebox</i> pada aplikasi <i>Debt Manager</i> telah dinyatakan lolos tanpa adanya <i>error</i> untuk fungsionalitas utamanya.	Penelitian sebelumnya menggunakan metode <i>whitebox</i> untuk menguji aplikasi <i>debt manager</i> , sedangkan penelitian yang akan dilakukan untuk menerapkan <i>whitebox testing</i> pada <i>rest-api</i> aplikasi desa bersinar.

No	Judul	Penulis	Tahun	Studi kasus	Metode / Algoritma	Hasil	Perbedaan dengan penelitian yang dilakukan
5	Analisis Perbandingan Performa Web Service Rest Menggunakan Framework Laravel, Django dan Node Js Pada Aplikasi Berbasis <i>Website</i>	Abi Amarulloh, kurniasih dan Muchlis	2023	Pengujian kecepatan <i>request</i> data	Framework Laravel, Django dan Node Js	Menampilkan performa web service yang tinggi pada aplikasi client berbasis <i>website</i> yaitu pada runtime NodeJs.	Pada penelitian sebelumnya terdapat pengujian kecepatan <i>request</i> data per detik pada framework Laravel, Django dan Node Js. Pada penelitian yang akan dilakukan penulis akan menggunakan NodeJs sebagai runtime environment pada ExpressJs.

2.2 Dasar Teori

2.2.1 Narkotika Psikotropika dan Obat Terlarang (NARKOBA)

Narkoba adalah singkatan dari narkotika, psikotropika dan obat terlarang yang merupakan zat adiktif yang sangat berbahaya jika dikonsumsi oleh manusia. Ketika manusia sudah mengonsumsi narkoba akan ada efek ketergantungan yang dapat menyebabkan penurunan atau perubahan kesadaran, mudah marah, mengantuk dan menyebabkan rasa malas. Dengan efek ketergantungan yang sangat berbahaya, manusia dapat melakukan suatu hal terlarang seperti mencuri atau menjual barang apapun hanya untuk mendapatkan narkoba. Penyalahgunaan narkoba dapat disebabkan oleh faktor internal maupun eksternal yang saling mempengaruhi dan memperkuat satu sama lain dalam mempengaruhi penyalahgunaan narkoba. Hukuman bagi pelaku penyalahgunaan narkoba adalah rehabilitasi dan sanksi sesuai dengan undang undang yang berlaku di Indonesia [7].

2.2.2 Badan Narkotika Nasional (BNN)

Badan Narkotika Nasional atau BNN merupakan sebuah lembaga pemerintah yang mempunyai tugas pencegahan, pemberantasan penyalahgunaan dan peredaran gelap narkotika, psikotropika dan bahan adiktif lainnya. BNN dipimpin oleh seorang kepala yang bertanggung jawab secara langsung kepada presiden. BNN diberikan kewenangan penyelidikan dan penyidikan tindak pidana narkotika menurut UU Nomer 35 Tahun 2009 [8]. BNN akan bersinergi dengan kepolisian dalam mengungkap dugaan tindak pidana penyalahgunaan narkoba yang diawali dari penyidikan, penangkapan, pengeledahan dan tindakan apapun yang dapat mencari titik terang dugaan penyalahgunaan narkoba. Dalam menangani penyalahgunaan narkoba, BNN mempunyai program yang dapat menekankan pengurangan penyalahgunaan narkoba seperti program Desa Bersih Narkoba (Desa bersinar).

Program Desa Bersinar adalah program dari Badan Narkotika Nasional (BNN) yang melibatkan partisipasi aktif dan komitmen dari perangkat desa hingga masyarakat desa. Kegiatan Desa Bersinar dilaksanakan pada satuan wilayah

setingkat kelurahan atau desa yang memiliki kriteria untuk melaksanakan program Pencegahan, Pemberantasan, Penyalahgunaan dan Peredaran Gelap Narkotika (P4GN) yang dilaksanakan secara massif. Tujuan program desa bersinar yaitu untuk memberikan pengayoman dan perlindungan dari gangguan ketenteraman serta ketertiban di desa agar terciptaya situasi yang aman, nyaman dan tentram. Kegiatan yang dilakukan untuk mewujudkan Desa Bersinar yaitu sebagai berikut:

1. Melaksanakan Komunikasi, Informasi, dan Edukasi (KIE) Pencegahan Penyalahgunaan Narkoba kepada seluruh unsur masyarakat:
 - a. KIE Individu atau kunjungan ke rumah sasaran.
 - b. KIE Kelompok yaitu memanfaatkan forum sosial seperti pengajian, arisan, dan kegiatan posyandu.
 - c. KIE Massa yaitu memanfaatkan media konvensional seperti hiburan rakyat dan pertunjukan seni.

2. Pembentukan Relawan Anti Narkoba dan Penggiat Anti Narkoba

Kepala desa membentuk relawan anti narkoba dari berbagai unsur masyarakat desa. Pembentukan relawan tersebut bertujuan untuk menggerakkan masyarakat desa agar berperan aktif dalam upaya P4GN. Sehingga Pembentukan relawan anti narkoba dapat memunculkan penggiat anti narkoba.

3. Mendorong terbentuknya Intervensi Berbasis Masyarakat (IBM)

IBM merupakan kegiatan yang dilakukan oleh masyarakat terhadap korban penyalahgunaan narkoba dengan memanfaatkan potensi pada sumber daya masyarakat. Kegiatan IBM dimulai dari pemetaan wilayah, penjangkauan, intervensi, pemantauan hingga pendampingan melalui pendekatan kearifan lokal.

Terdapat Indikator keterpullihan kawasan rawan (IKKR) narkoba dalam melakukan pemetaan kawasan narkoba. Kerawanan suatu daerah diukur dengan 8 indikator pokok dan 5 indikator pendukung. Untuk melakukan pengukuran keterpullihan terdapat 7 indeks keberhasilan program. Berikut merupakan bobot dari indikator pokok, indikator pendukung dan indikator keterpullihan [9]:

Tabel 2.2 Bobot Indikator

No	Jenis Indikator	Indikator	Bobot
1	Indikator Pokok Kerawanan Narkoba	Adanya bandar narkoba	8
2		Adanya produksi narkoba	7
3		Adanya entry point narkoba	6
4		Adanya kurir narkoba	5
5		Adanya pecandu narkoba	4
6		Adanya kasus narkoba	3
7		Adanya barang bukti narkoba	2
8		Adanya kejahatan umum	1
9	Indikator Pendukung	Terawasnya tempat hiburan	5
10		Menurunnya angka kemiskinan	4
11		Terawasnya kos - kosan	3
12		Bertambahnya sarana publik	2
13	Tingkat Keberhasilan Program	Meningkatkan interaksi sosial	1
14		Hasil yang dirasakan masyarakat	7
15		Keberlanjutan program	6
16		Pelaksana program	5
17		Pelaksana program	4
18		Sinergi instansi pemerintah	3
19		Kemitraan dunia usaha	2
20		Dampak keberdayaan Masyarakat	1
Jumlah Total (N = 20)			79

Untuk mengetahui perkembangan dari setiap indikator dan tingkat keberhasilan, pelaksana program dapat berkordinasi dengan aparat penegak hukum, seperti: polisi, jaksa, hakim, pabean dan bea cukai, petugas rutan/ lapas, media massa, pemerintah daerah, camat, lurah atau kepala desa serta melalui identifikasi [9]. Aparat penegak hukum dapat memberikan nilai antara 1 sampai 4 dari setiap pertanyaan yang terdapat pada formulir. Hasil dari *respons* tersebut dihitung dengan cara nilai dikali bobot dan jumlahkan hasil perkalian tersebut, selanjutnya dibagi dengan total jumlah bobot. Hasil dari perhitungan tersebut bernilai interval. Penentuan status bahaya, waspada, siaga dan aman pada suatu

daerah ditentukan dari nilai interval yang sudah dihitung. Berikut merupakan penentuan status suatu daerah berdasarkan nilai interval:

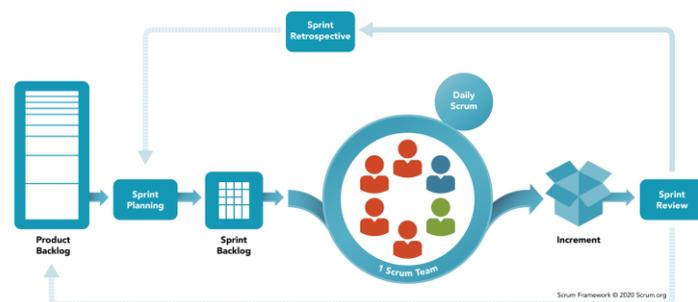
Tabel 2.3 Status Daerah Rawan Narkoba

Nilai Interval	Nilai Interval Konversi	Kategori IKKR	Kriteria Pulih
1,00 - 1,75	25,00 - 43,75	A	BAHAYA
1,76 - 2,50	43,76 - 62,50	B	WASPADA
2,51 - 3,25	62,51 - 81,25	C	SAIGA
3,26 - 4,00	81,26 - 100,00	D	AMAN

2.2.3 Scrum

Scrum merupakan kerangka kerja pada pengembangan perangkat lunak yang dilakukan secara berulang dengan mengidentifikasi penyebab masalah dan dilanjutkan berdiskusi dengan tim untuk mencari solusi dari setiap permasalahan [6]. Pengembangan perangkat lunak menggunakan metode *scrum* dapat diprediksi lebih cepat untuk mengetahui resiko dan kualitas perangkat lunak [10]. Terdapat tahapan dalam proses pengembangan perangkat lunak pada metode *scrum*, tahapan tersebut yaitu sebagai berikut:

SCRUM FRAMEWORK



Scrum.org

Gambar 2. 1 Metode *Scrum* [11]

1. *Product backlog*

Tahap *product backlog* adalah hal yang paling penting dalam pengembangan perangkat lunak menggunakan metode scrum. Pada tahap ini akan merubah ide menjadi suatu nilai yang nantinya akan dikerjakan untuk mencapai *product goal* [12]. Agar tahapan ini dapat mencapai *product goal*, maka pada tahap ini akan menentukan fitur yang akan dikembangkan dan durasi waktu agar tercipta konsisten selama pengembangan perangkat lunak.

2. *Sprint Planning*

Sprint Planning merupakan tahapan penting dalam memulai pekerjaan selama *sprint*. Pada tahap ini *product owner* akan melakukan rapat dengan anggota tim untuk memastikan *product backlog* yang akan dikerjakan dan membahas hubungan antara *product backlog* dengan *product goal* [12]. Hasil dari rapat tersebut tim developer akan memberikan estimasi waktu pengerjaan dari setiap *product backlog*.

3. *Daily Scrum*

Daily Scrum merupakan tahapan pada *sprint* yang dilakukan untuk memastikan implementasi *product backlog* terus berjalan sesuai dengan *product goal* [12]. Pada tahap ini tim akan melakukan rapat 15 menit setiap harinya untuk membahas kemajuan *product backlog* yang telah dikerjakan dan rencana kerja yang akan dilakukan. Dengan melakukan *daily scrum* dapat meningkatkan komunikasi antar tim, mengidentifikasi hambatan, dan mendorong pengambilan keputusan yang cepat ketika terjadi hambatan.

4. *Sprint Review*

Sprint Review adalah tahapan untuk mempresentasikan hasil dari *sprint* yang sudah dikerjakan kepada pemangku kepentingan [12]. Pada tahap ini setiap *product backlog* akan dikatakan selesai jika *product backlog* berhasil dalam pengujian fitur. Setelah *product backlog* dan fitur dilakukan pengujian maka akan diakumulasi apakah terdapat *product backlog* yang tidak berhasil. Apabila terdapat *product backlog* yang tidak berhasil maka *product backlog* akan dikerjakan pada *sprint* berikutnya.

5. *Sprint Restropective*

Sprint Restropective merupakan tahap yang dilakukan untuk merencanakan cara agar dapat meningkatkan kualitas dan efektivitas pada perangkat lunak yang sudah dibangun [12]. Pada tahap ini akan membahas apa saja yang berjalan dengan baik saat melakukan *sprint*, masalah yang ditemui, dan masalah tersebut terselesaikan atau tidak. Dengan adanya tahap ini dapat dijadikan evaluasi agar *sprint* berikutnya dapat berjalan lebih baik.

2.2.4 *Unifed Modeling Language (UML)*

Unifed Modeling Language atau disingkat UML adalah pemodelan dalam merancang perangkat lunak yang telah distandarisasi sebagai media penulisan *blueprint* [13]. *Blueprint* dalam UML didefinisikan sebagai penjelasan informasi yang lebih detail meliputi konsep proses bisnis, penulisan kelas – kelas dalam bahasa program yang spesifik, skema database dan komponen yang diperlukan pada perangkat lunak. Dalam pengembangan perangkat lunak diperlukan program UML sebagai program yang terus berkelanjutan yaitu program yang akan diimplementasikan dalam jangka waktu yang panjang. Penggunaan UML dapat mempermudah dan memenuhi kebutuhan pengguna dengan efisien, detail dan tepat, termasuk dalam faktor scalability dan security. Terdapat berbagai macam perancangan sistem menggunakan UML yaitu sebagai berikut:

1. *Use Case Diagram*

Pada *use case diagram* akan menggambarkan interaksi antara aktor dengan sistem [13]. Perilaku tersebut menjelaskan tipe *role* pengguna dengan sistem yang dapat dilakukan. Sehingga gambaran dari use case diagram dapat mempermudah *developer* dalam melakukan proses otorisasi pada sistem yang akan di kembangkan. Terdapat simbol dalam membuat use case diagram, diantaranya sebagai berikut:

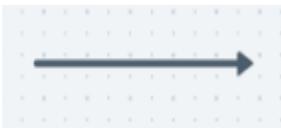
Tabel 2.4 Simbol *Use Case* Diagram

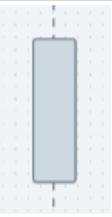
Simbol	Nama Simbol	Keterangan
	<i>Actor</i>	Menggambarkan interaksi antara manusia dengan sistem
	<i>Use case</i>	Menggambarkan interaksi antara sistem dengan <i>actor</i>
	<i>Association</i>	Menggambarkan penghubung antara <i>actor</i> dengan <i>use case</i>
	<i>Include</i>	Menggambarkan suatu <i>use case</i> merupakan fungsionalitas dari <i>use case</i> lainnya
	<i>Extend</i>	Menggambarkan suatu <i>use case</i> merupakan tambahan fungsionalitas dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

2. Sequence Diagram

Pada *Sequence* diagram akan menggambarkan proses interaksi antara objek - objek dalam sebuah sistem [14]. Pada tahap ini diagram akan digambarkan secara detail dari aksi yang dilakukan aktor sampai mendapatkan *respons*. Untuk menggambarkan aksi sampai mendapatkan *respons* perlu adanya objek - objek untuk menggambarkan proses yang terjadi pada suatu sistem. Berikut merupakan simbol dalam membuat *Sequence* diagram:

Tabel 2.5 Simbol *Sequence* Diagram

Simbol	Nama Simbol	Keterangan
	<i>Actor</i>	Menggambarkan interaksi antara manusia dengan sistem
	<i>Object</i>	Menggambarkan objek pada suatu sistem
	<i>Message</i>	Menggambarkan pengiriman suatu pesan
	<i>Replay Message</i>	Menggambarkan pengiriman Kembali suatu pesan
	<i>Message Return</i>	Menggambarkan balasan dari pengiriman suatu pesan

Simbol	Nama Simbol	Keterangan
	<i>A focus of control & alife line</i>	Menggambarkan mulai dan berakhirnya suatu <i>message</i>

2.2.5 Entity Relationship Diagram (ERD)

Entity Relationship Diagram atau disingkat ERD adalah diagram yang menggambarkan model perancangan suatu basis data yang memiliki entitas dan relasi antar entitas [15]. Didalam *Entity Relationship Diagram* terdapat tiga elemen dasar yaitu entitas, atribut dan relasi. Atribut dalam entitas merupakan pendeskripsian kegiatan yang dilakukan entitas dan merupakan karakteristik dari suatu entitas. Didalam entitas harus memiliki *primary key* sebagai ciri khas dari entitas.

2.2.6 Javascript

Javascript adalah bahasa pemrograman berbentuk kumpulan script yang berjalan pada suatu dokumen *Hyper Text Markup Language* (HTML) dengan mempunyai karakteristik berorientasi pada objek dan merupakan bahasa pemrograman berjenis *high-level* [pengertian javascript]. Javascript digunakan pada pengembangan perangkat lunak agar lebih dinamis dan interaktif serta dapat meningkatkan fungsionalitas dalam pengembangan perangkat lunak. Javascript dapat digunakan pada sisi client dan sisi server. Pada sisi client mesin javascript browser mengkonversi kode javascript menjadi *bytecode* untuk perantara antar sintaks javascript dengan mesin. Pada sisi server mesin javascript berada langsung di server untuk mengakses basis data, melakukan logika yang berbeda, dan memberikan *respons* berbagai event. Javascript merupakan bahasa pemrograman yang cukup kompleks dan sangat fleksibel, dapat digunakan untuk *Application Programming Interfaces (API)* dan menggunakan *framework* untuk pengembangan perangkat lunak dengan cepat.

2.2.7 Javascript Object Notation (JSON)

Javascript Object Notation atau disingkat *JSON* adalah suatu format data dari bahasa pemrograman *javascript*. *JSON* memiliki dua struktur dalam format data yaitu berbentuk *object* dan *array* [16]. Struktur yang berbentuk *object* diawali dengan kurung kurawal buka dan diakhiri dengan kurung kurawal tutup, sedangkan struktur yang berbentuk *array* diawali dengan tanda kurung siku buka dan diakhiri dengan kurung siku tutup. Nilai data dalam format *JSON* diawali dengan nama atribut berbentuk string selanjutnya terdapat titik dua dan berisi nilai dari suatu atribut.

2.2.8 ExpressJS

ExpressJS adalah framework dari *NodeJs* yang dirancang secara fleksibel dan sederhana untuk membantu tahap pengembangan aplikasi backend. *ExpressJs* memberikan kebebasan bagi developer untuk mendesain aplikasi, sehingga setiap developer memiliki rancangan arsitektur yang berbeda dalam membangun software. *ExpressJs* dapat menggunakan library yang dapat memudahkan developer dalam bekerja pada pembuatan token, session maupun autentikasi *login* yaitu menggunakan library *Json Web Token* (*JWT*). Framework *EkpressJs* memiliki kecepatan dan kemudahan dalam membangun *Representational State Transfer Application Programming Interface* (*Rest API*) yang nantinya digunakan dalam proses *create,read,update* dan *delete* (*CRUD*) data. *ExpressJS* bekerja dalam *runtime environment* *NodeJS* pada *V8 Chrome* yang di mana akan memudahkan pertukaran data dari server ke client dengan jaringan yang cepat dengan beban yang ditanggung lebih tinggi tanpa harus mengorbankan performa atau keuntungan dari sisi server, sehingga permintaan *CRUD* yang diterima oleh server dapat di proses secara langsung. Pemanfaatan *Restful* atau *Representational State Transfer* pada *ExpressJs* dapat digunakan untuk membangun berbagai macam *client-side* seperti pada *website, mobile* maupun *desktop*.

2.2.9 MySQL

MySQL adalah sebuah *Database Management System* (*DBMS*) yang mempunyai fungsi sebagai *Relational Database Management System* (*RDBMS*). *Database* ini menyimpan data dalam bentuk tabel yang terdiri dari baris dan

kolom dengan atribut data yang berbeda serta memiliki hubungan antar atribut data. Kinerja yang sangat cepat, reliable, dan mudah untuk digunakan serta bekerja dengan arsitektur client server atau embedded system [17]. Untuk melihat data yang ada pada *database MySQL* dapat membuka software *phpMyAdmin* atau *MySQL Workbench*. Pada pengembangan perangkat lunak untuk mengakses data terdapat oprasi dasar untuk memanipulasi data dengan cara terstruktur dan terstandarisasi, oprasi dasar tersebut yaitu sebagai berikut:

1. *Create*

Oprasi *create* digunakan untuk membuat atau menambahkan data baru ke dalam sistem. Dalam konteks *database*, data yang sudah ditambahkan akan menambahkan baris baru ke dalam tabel di database. Metode HTTP yang digunakan pada proses *create* yaitu menggunakan *POST*.

2. *Read*

Oprasi *read* digunakan untuk membaca atau mengambil data yang ada pada sistem. Dalam konteks *database*, *read* akan mengambil data dari suatu tabel di database dan biasanya akan digunakan untuk menampilkan data. Metode HTTP yang digunakan pada oprasi *read* yaitu menggunakan *GET*.

3. *Update*

Oprasi *Update* digunakan untuk mengedit suatu data yang ada pada sistem. Dalam konteks *database*, *update* akan merubah data pada baris table di tabase sesuai dengan id atau kode unik dari data yang akan di edit. Metode HTTP yang digunakan pada oprasi *update* yaitu menggunakan *PUT* atau *PATCH*.

4. *Delete*

Oprasi *Delete* digunakan untuk menghapus suatu data yang ada pada sistem. Dalam konteks *database*, *delete* akan menghapus data pada baris table di *database* sesuai dengan id atau kode unik dari data yang akan dihapus. Metode HTTP yang digunakan pada oprasi *delete* yaitu menggunakan *DELETE*.

2.2.10 *Sequelize Object Relation Mapping (ORM)*

Sequelize Object Relation Mapping (ORM) merupakan teknik dalam melakukan interaksi terhadap relational database atau dalam *RDMS*. Dengan menggunakan ORM pengembangan perangkat lunak akan lebih mudah dikarenakan untuk mekases database tidak diperlukan *query* manual yang cukup memakan waktu. *ORM Sequelize* digunakan pada runtime NodeJS berbasis *promise-based* yaitu cara yang lebih modern dan mudah untuk mengelola oprasi *asynchronous* [18]. *ORM Sequelize* merupakan suatu ORM pada nodeJS yang menerapkan *syntax Promise Javascript*. Fitur yang dapat digunakan *ORM Sequelize* yaitu proses membuat database, migration, seeding data, proses CRUD, validasi data dan hooks. Cara kerja *ORM Sequelize* pada proses CRUD yaitu mendefinisikan suatu object yaitu *findAll* untuk menampilkan data, *create* untuk menambahkan data, *update* untuk mengedit data dan *destroy* untuk menghapus data. *ORM Sequelize* dapat digunakan untuk database SQL seperti MySQL, PostgreSQL, MariaDB, SQLite dan Microsoft SQL server.

2.2.11 *White Box*

Whitebox adalah metode pengujian perangkat lunak dengan cara menganalisa dan meneliti stuktur internal dan kode dari perangkat lunak [19]. Pengujian *whitebox* dilakukan untuk mengetahui sistem dapat bekerja secara struktural dan mengetahui kode program yang dikerjakan apakah terdapat kesalahan [20]. Dalam pengujian menggunakan *whitebox* akan terdapat *flowchart*, *flowgraph*, *cycleomatic complexity*, *independet Path* dan uji skenario. Perhitungan *cycleomatic complexity* yaitu dengan menghitung jumlah *edges* pada *flowgraph* dan jumlah *nodes* pada *flowgraph*. Hasil dari jumlah *edges* dikurangkan dengan jumlah *nodes* dan ditambahkan dengan nilai 2. Dari hasil tersebut *indipendent patah* merupakan jumlah jalur yang dapat dilakukan untuk melakukan testing.

Tabel 2.6 Tingkat Resiko Kode Program

Path	Tipe Prosedur	Tingkat Resiko
1-4	Sederhana	Rendah
5-10	Struktur dan Stabil	Rendah
11-20	Kompleks	Menengah
21-50	Kompleks dan Kritis	Tinggi
>50	Rentan kesalahan, mengganggu, tidak dapat diuji	Sangat Tinggi

2.2.12 *Black Box*

Black box testing merupakan salah satu metode yang digunakan untuk melakukan pengujian fungsionalitas dengan melihat hasil eksekusi melalui data uji dan memastikan fungsi dari software. Pengujian menggunakan *blackbox testing* dapat mengetahui masukan dan keluaran yang di masukan oleh user dapat berjalan dengan baik atau tidak, dan mengevaluasi kesesuaian dengan kebutuhan pengguna tanpa mengetahui struktur kode atau program.