

BAB II

TINJAUAN PUSTAKA

2.1 Teori Terkait

Berdasarkan penelitian yang telah dirancang dan dikerjakan oleh Hendri, Rezky Nanda Saputra dan Desy Kisbianty yang berjudul Perancangan Aplikasi Rencana Anggaran Biaya Pada PT. Zaki Putra Andalas Jambi. Penelitian ini bertujuan untuk menyajikan solusi melalui pengembangan aplikasi pengelolaan rencana anggaran biaya yang dibangun menggunakan *tools* bahasa pemrograman PHP serta basis data MySQL, sehingga proses pengolahan data menjadi lebih terorganisir dan mampu menyajikan informasi secara cepat serta akurat. Pendekatan yang diterapkan dalam proses pengembangan sistem ini ialah model waterfall, dengan pemodelan sistem yang digambarkan menggunakan *use case diagram* dan *flowchart*. Sebuah aplikasi yang dihasilkan mampu menyediakan informasi berupa data pekerjaan, data pelanggan, analisis bahan dan upah, serta laporan rencana anggaran biaya.[3].

Penelitian yang sama juga dilakukan oleh Sarwido dan Dony Hermawan yang berjudul Perancangan dan Pengembangan Sistem Informasi Berbasis Android untuk Rencana Anggaran Biaya Rumah Minimalis di Kabupaten Jepara tahun 2022. Pengembangan aplikasi ini menggunakan metode *Rapid Application Development* (RAD) karena dianggap lebih efisien dalam proses pembangunan sistem. Aplikasi tersebut dibuat dengan memanfaatkan software Android Studio serta bahasa pemrograman Kotlin.

sistem informasi RAB berbasis Android yang Aplikasi ini memberikan manfaat bagi masyarakat umum sebagai bahan referensi dalam menentukan estimasi anggaran pembangunan rumah minimalis tipe 21, 36, dan 45. Melalui sistem ini, pengguna dapat melihat hasil perhitungan biaya kapan saja dan di mana saja, serta memiliki kemampuan untuk menyesuaikan harga satuan sesuai kebutuhan sehingga dapat mengetahui total biaya yang diperlukan secara lebih tepat. Penerapan sistem ini mempermudah masyarakat dalam memperkirakan biaya pembangunan rumah minimalis, serta menyajikan informasi perhitungan yang lebih selektif, relevan, dan akurat. Berdasarkan hasil pengujian yang dilakukan oleh ahli materi dengan persentase kelayakan 100%, ahli media sebesar 86,6%, serta angket responden dengan persentase 90,6%, dapat disimpulkan bahwa aplikasi informasi perhitungan RAB rumah minimalis ini dinilai sangat layak digunakan sebagai sumber informasi dalam proses pembangunan rumah minimalis [4].

Penelitian lain juga dilakukan oleh Sawindri dan Agung Tri Ayudi yang berjudul sistem informasi perhitungan rencana anggaran biaya perencanaan bangunan yang dikembangkan menggunakan metode *framework codeigniter 3* pada tahun 2022. Aplikasi ini dirancang dan dibangun dengan memanfaatkan *framework codeigniter 3* sebagai dasar pengembangan sistem. Adapun bahasa pemrograman yang digunakan adalah *Hypertext Preprocessor* (PHP), sementara untuk pengelolaan basis datanya memanfaatkan XAMPP melalui phpMyAdmin sebagai media penyimpanan.

Framework codeigniter 3 merupakan salah satu kerangka kerja dalam pengembangan sistem informasi berbasis web yang menyediakan kumpulan class, function, serta konsep yang telah ditata secara terstruktur berdasarkan arsitektur model view controller (MVC). Aplikasi RAB ini dikembangkan untuk mendata dan menghitung besaran anggaran biaya serta memperkirakan durasi penyelesaian pekerjaan sesuai dengan target pelaksana. Aplikasi perhitungan ini secara khusus digunakan untuk menghitung estimasi “Perencanaan Desain Berulang Pembangunan Rumah Susun Tahun 2019”. Pengujian sistem dilakukan menggunakan metode black-box, yang bertujuan memastikan bahwa aplikasi berjalan dengan baik dan mengidentifikasi kendala pada penggunaan fitur yang tersedia[5].

Perancangan Sistem Informasi Akuntansi RAB pemeliharaan peralatan menggunakan Microsoft Visual Studio 2010 dan penyimpanan *database* MySQL, berupa studi deskriptif pada Dinas ATR/BPN Kota Bandung tahun 2023. Sistem ini dikembangkan dengan MS Visual Studio 2010 dan MySQL sebagai basis datanya. Rencana Anggaran Biaya (RAB) menjadi atribut penting untuk mengetahui total anggaran dalam satu periode akuntansi. Pada tahap realisasi, pemanfaatan sistem tersebut pengelolaan anggaran RAB harus dikontrol dengan terus-menerus agar selalu sesuai dengan apa yang diharapkan pada saat penyusunan RAB tersebut. RAB yang tergabung dengan sistem pantatan umum akan menyebabkan tidak sesuaian data dan harus melekukan revisi secara berulang kali. Subuah sistem terpisah dengan

kemampuan setara dengan sistem utama akan sangat membantu pengolahan data RAB [6].

Penelitian terkait juga pernah dilakukan oleh Siti Umi Damayanti dan Jumrianto dengan judul Desain Sistem Informasi Berbasis Website, Monitoring RAB Menggunakan Bahasa Pemrograman HTML dan PHP pada tahun 2022. Studi tersebut mengembangkan sistem informasi berbasis web untuk memantau RAB dengan memanfaatkan HTML dan PHP. Sistem ini digunakan untuk mendukung pengelolaan transaksi harian serta membantu proses operasional dan strategis organisasi. Dalam pengembangannya, penulis menerapkan metode perancangan sistem dan memvisualisasikan alurnya melalui *flowchart* serta *use case diagram*. Sistem yang dibuat bertujuan mempermudah proses monitoring RAB serta mengurangi risiko hilangnya dokumen atau berkas terkait[7].

Berdasarkan hasil telaah terhadap penelitian sebelumnya, dapat diketahui bahwa terdapat sejumlah perbedaan pada sistem informasi Rencana Anggaran Biaya (RAB) berbasis web yang di CV. Media Pratama Solusinet, salah satunya terletak pada penggunaan *tools* yang lebih mutakhir.

2.2 Landasan Teori

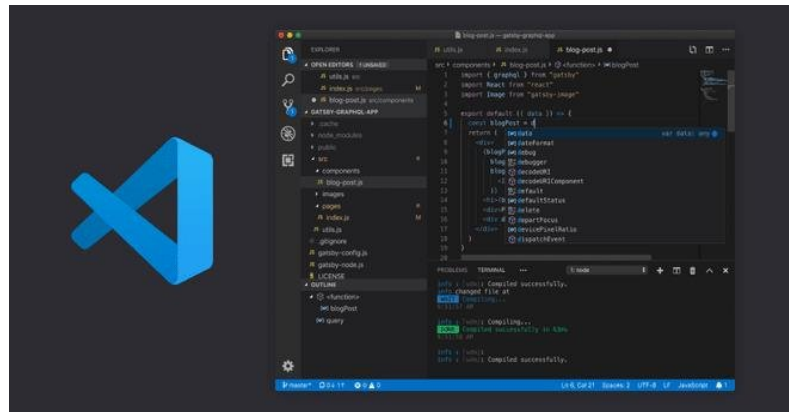
2.2.1 Rencana Anggaran Biaya

Rencana Anggaran Biaya (RAB) adalah perkiraan atau proyeksi biaya yang diprediksi akan diperlukan dalam pelaksanaan suatu proyek tertentu. Di dalam penyusunan RAB, terdapat komponen

perhitungan yang meliputi kebutuhan material hingga jumlah tenaga kerja yang dibutuhkan untuk menyelesaikan pekerjaan tersebut. RAB adalah perhitungan anggaran yang mencakup biaya bahan, upah, serta berbagai pengeluaran lain yang terkait dengan pelaksanaan suatu proyek. RAB adalah estimasi pengeluaran yang mencakup biaya material serta tenaga kerja, yang digunakan sebagai dasar pertimbangan dalam proses penawaran suatu proyek.

2.2.2 Visual Studio Code

Visual Studio Code (VS Code) adalah *editor* teks yang ringan namun memiliki performa tinggi, yang dikembangkan oleh Microsoft dan dapat digunakan pada berbagai sistem operasi seperti Linux, macOS, maupun Windows. *Editor* ini mendukung bahasa pemrograman JavaScript, TypeScript, dan Node.js, serta dapat diperluas untuk bahasa lain melalui pemasangan plugin di marketplace VS Code, seperti C++, C#, Python, Go, Java, dan beberapa bahasa pemrograman tambahan. VS Code menyediakan beragam fitur, antara lain *intellisense*, integrasi *git*, kemampuan debugging, serta dukungan ekstensi untuk meningkatkan fungsionalitas *editor*. Fitur-fitur ini terus berkembang sejalan dengan pembaruan versi VS Code yang dirilis secara rutin menjadi salah satu keunggulan dibandingkan *editor* teks lainnya.



Gambar 2.1 Visual Studio Code

Visual Studio Code atau VSCode adalah *editor* teks yang ringan tetapi memiliki performa baik, dikembangkan oleh Microsoft dan dapat digunakan pada berbagai platform. *Editor* ini memiliki dukungan bawaan untuk bahasa pemrograman seperti JavaScript dan TypeScript, serta dapat diperluas untuk bahasa lain seperti HTML, C++, C#, *Go*, *Java*, dan sebagainya. Aplikasi ini kompatibel dengan beragam sistem operasi, termasuk Linux, macOS, dan Windows [8].

2.2.3 Laravel

Laravel merupakan salah satu kerangka kerja berbasis PHP yang bersifat *open-source* dan dirancang dengan beragam modul inti untuk mendukung optimalisasi pengembangan aplikasi *web*. Peran *Laravel* menjadi penting karena PHP sebagai bahasa pemrograman memiliki sifat dinamis, sehingga pemanfaatannya dapat mempercepat proses pengembangan, meningkatkan aspek keamanan, serta menyederhanakan implementasi.

Laravel adalah framework untuk aplikasi berbasis PHP yang menyediakan ekosistem lengkap, didukung oleh berbagai paket dan ekstensi yang kompatibel. Dalam beberapa tahun terakhir, *laravel* telah mengalami kemajuan yang pesat. Hal ini tercermin dari meningkatnya minat para pengembang untuk memanfaatkan framework ini. *Framework laravel* dipilih karena mampu menyederhanakan proses pengembangan aplikasi secara signifikan..

Laravel pertama kali dirilis oleh Taylor Otwell pada 9 Juni 2011. Pada awal pengembangannya, *Laravel* diciptakan sebagai alternatif untuk *codeigniter*, karena Otwell menilai framework tersebut belum menyediakan sejumlah fitur penting, seperti dukungan autentikasi dan otorisasi secara bawaan. Sejak peluncuran awal, *laravel* telah mengalami berbagai peningkatan melalui pembaruan dan penambahan fitur. Hingga saat ini, *framework* tersebut telah mencapai versi ke-12 dengan berbagai keunggulan yang mendukung pengembangan aplikasi web modern.



Gambar 2.2 Laravel

2.2.4 XAMPP

XAMPP merupakan hasil pengembangan dari tim proyek yang dikenal dengan nama *Apache Friends*, yang di dalamnya terdapat tiga kelompok utama dalam tim, yaitu Tim Inti (*Core Team*), Tim Pengembang, dan Tim Pendukung. XAMPP merupakan akronim yang masing-masing hurufnya memiliki makna tertentu. X menunjukkan bahwa aplikasi ini kompatibel dengan berbagai sistem operasi, seperti *Windows*, *Linux*, *macOS*, serta *Solaris*. A=Apache, merupakan aplikasi *webserver*. Tugas utama *apache* adalah menghasilkan halaman *web* yang benar kepada *user* berdasarkan kode PHP yang dituliskan oleh pembuat halaman *web*.

Dalam beberapa kasus, perintah PHP dapat membutuhkan akses ke database seperti MySQL, untuk menghasilkan halaman *web* M adalah MySQL, yaitu *server* basis data yang menggunakan SQL sebagai bahasa pengelola data. MySQL memungkinkan pembuatan serta pengaturan isi *database*, termasuk menambah, mengubah, dan menghapus data. P merujuk pada PHP, bahasa pemrograman yang digunakan dalam pembuatan situs web.



Gambar 2.3 XAMPP

2.2.5 Database

Basis data atau *database* merupakan sistem yang berfungsi untuk menyimpan data, mengambil informasi sesuai kebutuhan, serta mengelola data yang tersimpan.

Database Management System (DBMS) adalah perangkat lunak yang berfungsi untuk membangun, mengelola, dan mengakses basis data secara efektif serta tersusun. Sementara itu, RDBMS merupakan jenis DBMS yang mendukung adanya hubungan antar data dalam bentuk relasional.

2.2.6 Flowchart

Flowchart merupakan *diagram* yang menggunakan simbol tertentu untuk menampilkan urutan proses secara rinci serta menunjukkan keterkaitan antara suatu instruksi dengan instruksi lainnya dalam sebuah program. *Flowchart* biasanya dimanfaatkan sebagai dokumen penjelas untuk menggambarkan alur logika sistem yang akan dibangun dan diserahkan kepada *programmer*, sehingga dapat membantu dalam menemukan solusi ketika terjadi permasalahan selama proses pengembangan sistem.

2.2.7 Bootstrap

Bootstrap merupakan sebuah *framework* pengembangan web yang memanfaatkan HTML, CSS, serta JavaScript, dan dikembangkan untuk mempercepat pembuatan tampilan situs yang responsif serta berorientasi pada perangkat seluler terlebih dahulu.



Gambar 2.4 Bootstrap

2.2.8 Hosting

Hosting layanan berbasis internet yang digunakan untuk menyimpan dan mengelola data situs atau aplikasi *web* agar dapat diakses secara online. *File* yang tersimpan dapat berupa gambar, skrip, *video*, *email*, aplikasi, maupun *database*. Untuk membuka sebuah situs, pengguna biasanya memasukkan nama domain atau kata kunci tertentu pada mesin pencari seperti Google.

2.2.9 Domain

Domain merupakan alamat yang menjadi identitas sebuah *website* atau komputer sehingga memudahkan proses akses. Pada umumnya, alamat situs diakhiri dengan ekstensi seperti *.com* atau *.net*. Tanpa penggunaan domain, pengguna harus mengetik deretan angka berupa *IP address* pada *browser* setiap kali mengunjungi situs, yang tentu menyulitkan karena jumlah *website* di dunia sangat banyak dan masing-masing memiliki *IP address* berbeda.

2.2.10 UML (Unified Modeling Language)

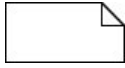
Unified Modeling Language merupakan salah satu pendekatan dalam rekayasa perangkat lunak yang digunakan untuk memvisualisasikan alur kerja sistem, fungsi, tujuan, serta mekanisme pengendalian yang ada. Dalam analisis dan perancangan sistem informasi, pendekatan yang umum digunakan saat ini adalah kombinasi konsep pemrograman yang berorientasi objek dan teknik pengembangan perangkat lunak, di mana setiap sistem dipandang sebagai objek mandiri, mencakup data serta prosesnya sendiri, dan dapat beroperasi secara utuh dalam satu paket sistem.

1. *Use Case Diagram* berfungsi untuk menggambarkan interaksi antara manusia (aktor) dengan fungsionalitas yang disediakan oleh sistem. Setiap use case dapat merepresentasikan berbagai jalur interaksi pengguna dengan sistem, di mana setiap jalur tersebut dikenal sebagai skenario.

Tabel 2.1 Simbol *Use Case Diagram*

No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang

No	Gambar	Nama	Keterangan
			tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>use case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (<i>sinergi</i>).

No	Gambar	Nama	Keterangan
10		Note	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

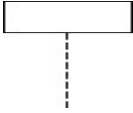
2. *Activity Diagram* adalah diagram yang menggambarkan alur kerja atau aktivitas suatu sistem dalam perangkat lunak.

Tabel 2. 2 Simbol *Activity Diagram*

No	Gambar	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

3. *Sequence Diagram* menggambarkan objek-objek yang terlibat dalam suatu use case serta aliran pesan yang terjadi antar objek tersebut. Diagram ini bersifat dinamis dan menekankan aktivitas objek berdasarkan urutan waktu terjadinya.

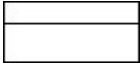
Tabel 2. 3 Simbol *Sequence Diagram*

No	Gambar	Nama	Keterangan
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

4. *Class Diagram* merupakan model pemodelan bersifat statis yang digunakan untuk mendeskripsikan elemen data serta aliran informasi dalam suatu sistem secara menyeluruh. Diagram ini biasanya berkaitan dengan rancangan struktur basis data atau dapat diterapkan sebagai alternatif ERD dalam proses pemodelan diagram pada rekayasa perangkat lunak secara tradisional.

Tabel 2. 4 Simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1		<i>Generalization</i>	Hubungan ini muncul ketika objek turunan (<i>descendent</i>) mewarisi perilaku dan struktur data dari objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Konsep ini digunakan sebagai langkah untuk menghindari asosiasi dengan lebih dari 2 objek.

No	Gambar	Nama	Keterangan
3		<i>Class</i>	Sekumpulan objek yang memiliki kesamaan dalam atribut dan operasi yang dimilikinya.
4		<i>Collaboration</i>	Penjelasan mengenai rangkaian tindakan yang ditampilkan oleh sistem untuk menghasilkan keluaran yang dapat diukur bagi seorang aktor.
5		<i>Realization</i>	Aktivitas yang secara langsung dieksekusi oleh sebuah objek.
6		<i>Dependency</i>	Relasi ini terjadi ketika adanya perubahan pada suatu elemen yang bersifat mandiri (<i>independent</i>) akan berdampak pada elemen lain yang bergantung padanya dan tidak bersifat mandiri
7		<i>Association</i>	Suatu elemen yang menjadi penghubung antara satu objek dengan objek lainnya.