

BAB II

Tinjauan Pustaka Dan Dasar Teori

2.1 Tinjauan Pustaka

Pada penelitian ini diperlukan beberapa referensi berupa penelitian-penelitian dengan kasus serupa. Adapun referensi penelitian terdahulu yang memiliki kesamaan dengan penelitian ini sebagai berikut:

Penelitian Gunawan, dkk. (2015) berfokus pada pengembangan sistem informasi yang mengelola data rumah kos, mulai dari alamat, fasilitas, hingga status ketersediaan kamar. Sistem ini membantu calon penyewa untuk mengetahui detail informasi terkait rumah kos yang mereka cari tanpa perlu melakukan survei langsung. Penelitian ini memberikan dasar yang kuat dalam pengelolaan informasi rumah kos secara digital, meskipun belum menyertakan fitur pemesanan *online*.

Selanjutnya, Budiman, dkk. (2019) memperluas fitur dengan pengembangan sistem informasi yang memungkinkan calon penyewa untuk membandingkan lokasi, fasilitas, dan harga rumah kos yang tersedia. Selain itu, fitur pemesanan *online* menjadi salah satu keunggulan dalam penelitian ini, di mana calon penyewa dapat memesan kamar kos secara langsung melalui platform web, sehingga proses pencarian dan penyewaan menjadi lebih efisien.

Penelitian Jannah, dkk. (2020) berfokus pada pengembangan sistem yang memungkinkan calon penyewa melakukan pemesanan kamar kos secara *online* dengan pilihan metode pembayaran seperti transfer bank atau COD. Selain itu, sistem ini memberikan fleksibilitas bagi pemilik kos untuk memperbarui data terkait kos secara *real-time*, seperti perubahan harga dan ketersediaan kamar.

Penelitian ini menawarkan peningkatan dalam aspek manajemen kos dari sisi pemilik, sehingga informasi yang diberikan kepada calon penyewa lebih akurat dan *up-to-date*.

Yusma, dkk. (2021) mengembangkan sistem informasi pencarian rumah kos berbasis web dengan metode *waterfall*, yang memberikan akses mudah bagi calon penyewa untuk mencari kos berdasarkan kriteria yang ditentukan. Sistem ini juga membantu pemilik kos untuk mempromosikan kosnya secara lebih luas melalui platform *online*, menghemat waktu dan biaya dalam proses pencarian dan penyewaan.

Penelitian Arimbi, dkk. (2022) berfokus pada pengembangan sistem berbasis *Laravel* dan *MySQL* untuk Kos Putri Malika, di mana proses pendaftaran dan reservasi kos dapat dilakukan secara *online*. Sistem ini dirancang untuk mempermudah calon penghuni dalam melakukan registrasi serta mencatat data penghuni *kos*. Dengan pendekatan *System Development Life Cycle (SDLC)*, penelitian ini menawarkan alur yang jelas mulai dari perencanaan hingga pengujian sistem.

Abdillah (2025) melengkapi penelitian sebelumnya dengan menambahkan fitur yang lebih kompleks. Sistem informasi yang dikembangkan menampilkan informasi detail tentang rumah kos, seperti nama kos, tipe kos, harga sewa bulanan, kategori sewa, peraturan kos, spesifikasi kamar, fasilitas umum, kamar mandi, parkir, hingga fasilitas di sekitar kos. Sistem ini juga mengintegrasikan *Google Maps* untuk menampilkan peta sesuai dengan alamat kos. Fitur pemilihan tanggal

masuk dan lama sewa serta metode pembayaran *online* menjadikan sistem ini lebih lengkap dan efisien dibandingkan penelitian sebelumnya.

Tabel 2. 1 Tabel Perbandingan

NO	Penulis	Judul Penelitian	<i>Tools</i> pengembangan	Fitur
1	Gunawan, dkk (2015)	Sistem Informasi Sewa Rumah Kos Dan Rumah Kontrakan Berbasis Web Di Surakarta	PHP, <i>MySQL</i>	Pencarian kos, harga kos, fasilitas kos, pemesanan rumah kos
2	Budiman, dkk (2019)	Perancangan Sistem Informasi Pencarian Dan Pemesanan Rumah Kos Berbasis Web	PHP, <i>MySQL</i>	Pencarian kos, harga kos, fasilitas kos, jenis kos, waktu sewa pemesanan kamar kos
3	Jannah, dkk (2020)	Sistem Informasi Pemasaran Rumah Kos Berbasis Web	PHP, <i>MySQL</i>	Pencarian kos, harga kos, fasilitas kos, jenis kos, kamar mandi, kamar dan dapat melakukan pemesanan kamar kos
4.	Yusma, dkk (2021)	Sistem Informasi Pencarian Rumah Kos Berbasis Web	PHP, <i>MySQL</i>	Pencarian kamar kos, Pemesanan kamar kos, menampilkan alamat rumah kos menggunakan <i>google maps</i> , menampilkan fasilitas dan foto kamar ,harga.

Lanjutan Tabel 2.1 Perbandingan

NO	Penulis	Judul Penelitian	<i>Tools</i> pengembangan	Fitur
5.	Arimbi, dkk (2022)	Rancangan Sistem Informasi Kos Putri Malika Berbasis Website Menggunakan Framework Laravel Dan Mysql	<i>PHP,MySQL</i>	Pencarian kamar kos, Pemesanan kamar kos, melakukan pembayaran secara <i>online</i> ,
	Abdillah (2025)	Sistem Informasi Rumah Kos Berbasis Web	<i>PHP,MySQL</i>	Pencarian kos, harga kos, jenis kos, fasilitas umum, fasilitas kamar, fasilitas kamar mandi, peraturan kamar, tempat terdekat dari kos, <i>Google maps</i> untuk menampilkan peta dan dapat melakukan pemesanan kamar kos

2.2 Dasar teori

2.2.1 Sistem informasi rumah kos

Sistem informasi merupakan gabungan dari beberapa elemen yang saling berkaitan satu sama lain yang membentuk satu kesatuan di dalam suatu organisasi

yang mengumpulkan, memproses, menyimpan dan mendistribusikan informasi sebagai pendukung pengambilan keputusan (Oktaviani dkk., 2019).

Menurut (Siregar, 2018), "Rumah kos adalah tempat tinggal sementara yang disewakan oleh pemilik kepada penyewa, biasanya terdiri dari beberapa kamar yang disewakan secara terpisah, dengan pembayaran bulanan atau tahunan."

Demikian dapat di simpulkan Sistem Informasi Rumah Kos merupakan sistem yang dapat mengumpulkan, memproses, menyimpan informasi seputar rumah kos untuk tujuan tertentu.

2.2.2 XAMP

Menurut (Nugroho, 2013) *XAMPP* adalah paket program web lengkap yang dapat dipakai untuk belajar pemrograman web, khususnya *PHP* dan *MySQL*.. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP*. Bagian penting dari *XAMPP* yang diasakan digunakan:

1. *Htdoc* adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas *PHP*, *HTML* dan *script* lain.
2. *Phpmyadmin* merupakan bagian untuk mengelola basis data *mysql* yang terdapat dikomputer. Untuk membukanya, buka browser lalu ketikkan alamat *http://localhost/phpmyadmin*, maka akan muncul halaman *phpmyadmin*.
3. Kontrol Panel yang berfungsi untuk mengelola layanan (*service*) *XAMPP*. Seperti menghentikan (*Stop*) layanan, ataupun memulai (*Start*).

Kegunaan *XAMPP* dari pengembangan sistem informasi ini adalah untuk menyediakan server web lokal yang dapat dijalankan di komputer. Ini memungkinkan untuk mengembangkan dan menguji aplikasi web secara lokal sebelum mempublikasikannya ke server yang sebenarnya.

2.2.3 PHP (Hypertext Preprocessor)

Menurut (Nugroho, 2013) *PHP (Hypertext Preprocessor)* itu bahasa pemrograman berbasis web. Jadi, *PHP* adalah bahasa program yang digunakan untuk membuat aplikasi berbasis web. *PHP* termasuk bahasa program yang bisa berjalan di sisi server, atau sering disebut *Side Server Language*. Jadi, program yang dibuat dengan kode *PHP* tidak bisa berjalan kecuali dia dijalankan pada server web, tanpa adanya server web yang terus berjalan dia tidak akan bisa dijalankan.

Sedangkan, Menurut (Kadir, 2013) *PHP* merupakan bahasa pemrograman yang ditunjuk untuk membuat aplikasi web. Ditinjau dari pemrosesannya, *PHP* tergolong berbasis server *side*. Artinya, pemrosesan dilakukan di server. Hal ini berkebalikan dengan bahasa seperti *JavaScript*, yang pemrosesannya dilakukan di sisi klien (*client side*).

Kegunaan *PHP* untuk pengembangan sistem informasi ini adalah *php* versi 8 untuk membawa perbaikan signifikan dalam kinerja eksekusi kode, yang dapat menghasilkan situs web yang lebih cepat dan responsif. Peningkatan kinerja ini dapat mempercepat waktu pemuatan halaman dengan versi terbaru dari *php 8* dapat memastikan bahwa aplikasi web yang di buat mendapatkan manfaat dari perbaikan keamanan terkini, membantu melindungi situs web dari potensi kerentanannya dan *PHP 8* juga memperkenalkan *Nullsafe Operator*, yang membantu dalam menangani

nilai *null* dengan lebih aman dan mengurangi risiko kesalahan terkait dengan nilai *null*.

2.2.4 MySQL

Database Manajemen Sistem adalah sebuah perangkat lunak yang digunakan untuk mengelola dan memanggil *query* dalam suatu basis data di mana data yang dikelola mampu menjadi informasi yang diinginkan dan dapat dilakukan proses pengambilan, penghapusan, pembaruan terhadap data secara mudah dan cepat.

MySQL adalah sebuah *RDBMS (Relational Database Management System)* digunakan sebagai sistem pengolahan data. Sistem *Database* ini memiliki dua bahasa yang digunakan untuk mengolah *database* yakni *Data Definition Language (DDL)* dan *Data Manipulation Language (DML)*. *DDL* digunakan untuk memanipulasi tabel sedangkan *DML* digunakan untuk memanipulasi data (Sibero, 2013). Dengan manajemen yang bersifat *relational* ini, data akan diletakkan pada beberapa tabel yang terpisah sehingga dalam pengelolaan data akan jauh lebih cepat.

Kegunaan *MySQL* dari pengembangan sistem informasi ini adalah untuk menyediakan platform untuk membuat, mengelola, dan menyimpan data, *MySQL* digunakan untuk menyimpan informasi seperti detail pengguna, konten situs, log transaksi, dan banyak lagi.

2.2.5 CodeIgniter

CodeIgniter merupakan *framework PHP* yang dibuat berdasarkan model *View Controlleer (MVC)*. *CI* memiliki *library* yang lengkap untuk mengerjakan operasi-operasi yang umum dibutuhkan oleh aplikasi berbasis web misalnya mengakses *database*, memvalidasi *form* sehingga sistem yang dikembangkan mudah (Basuki, 2010). *CI* juga menjadi satu-satunya *Framework* dengan dokumentasi yang lengkap dan jelas. *Source code* *CI* yang dilengkapi dengan *comment* didalamnya sehingga lebih memperjelas fungsi sebuah kode program dan *CI* yang dihasilkan sangat Bersih (*clean*) dan *search Engine Friendly (SEF)*. *Codeigniter* juga dapat memudahkan developer dalam membuat aplikasi web berbasis *PHP*, karena *framework* sudah memiliki kerangka kerja sehingga tidak perlu menulis semua kode program dari awal. Selain itu, struktur dan susunan logis dari *codeigniter* membuat aplikasi menjadi semakin teratur dan dapat fokus pada fitur-fitur apa yang akan dibutuhkan dalam pembuatan aplikasi tersebut.

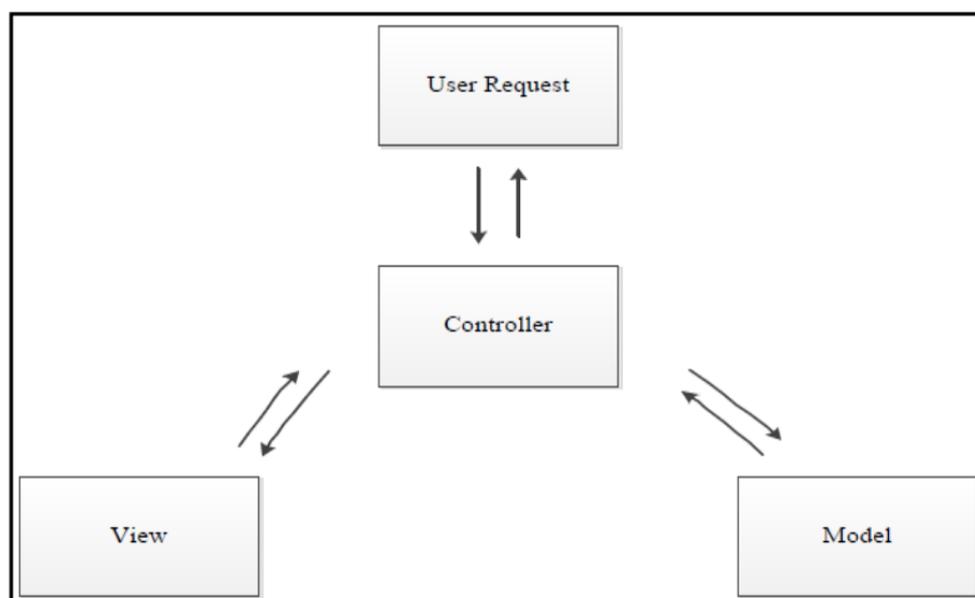
Codeigniter digunakan dalam pengembangan sistem informasi ini untuk mempercepat dan menyederhanakan proses pengembangan aplikasi web menggunakan bahasa pemrograman *PHP*, *CodeIgniter* mengikuti pola desain *Model-View-Controller (MVC)*, yang memisahkan logika aplikasi menjadi tiga komponen utama: Model (logika bisnis dan pengelolaan data), *View* (tampilan atau antarmuka pengguna), dan *Controller* (pengatur alur logika aplikasi). Struktur ini membantu dalam mengorganisir dan memisahkan kode, meningkatkan keterbacaan, dan memudahkan pemeliharaan. Tidak butuh banyak sumber daya

untuk menjalankan *Codeigniter*. Cukup dengan *text editor*, aplikasi *local server* dan file *Codeigniter*, sudah bisa menjalankan sebuah aplikasi *web* secara lokal. File *Codeigniter* sendiri berukuran kecil, yaitu hanya sekitar 2 MB, sehingga tidak banyak memakan kapasitas penyimpanan.

2.2.6 Model-View-Controller (MVC)

MVC merupakan konsep yang memisahkan data (Model) dari tampilan (View) dan cara bagaimana memprosesnya (Controller). Dengan konsep ini, pengembang dapat bekerja secara terpisah yakni mengerjakan bagian tampilan dan mengerjakan bagian *logic*.

Model merupakan komponen yang berguna untuk mengatur data, fungsi dan aturan dari aplikasi. View berfungsi untuk mengatur tampilan suatu aplikasi seperti teks, gambar, dll. Sedangkan controller merupakan komponen yang menghubungkan antara model dan *view* untuk menjalankan perintah tertentu.



Gambar 2. 1 Kosep Dasar MVC

Gambar 2.1 menerangkan bahwa ketika datang sebuah *user request*, maka akan ditangani oleh *controller*, kemudian *controller* akan memanggil model jika memang diperlukan operasi *database*. Hasil dari *query* oleh model akan dikembalikan ke *controller*. Setelah itu *controller* akan memanggil *view* yang tepat dan mengombinasikannya dengan hasil *query* model. Hasil akhir dari operasi ini akan ditampilkan di browser (Basuki, 2010). Dalam konteks *codeigniter* dan aplikasi berbasis web, maka penerapan konsep *MVC* mengakibatkan kode program dapat dibagi menjadi tiga kategori, yaitu:

1. *View*, merupakan bagian yang menangani *presentation logic*. Pada suatu aplikasi web bagian ini biasanya berupa *file template HTML*, yang diatur oleh *controller*. *View* berfungsi untuk menerima dan merepresentasikan data kepada *user*. Bagian ini tidak memiliki akses langsung terhadap bagian model.
2. Model, biasanya berhubungan langsung dengan *database* untuk memanipulasi data (*insert, update, delete, create*), menangani validasi dari bagian *controller*, namun tidak dapat berhubungan langsung dengan bagian *view*.
3. *Controller*, merupakan bagian yang mengatur hubungan antara bagian model dan bagian *view*, *controller* berfungsi untuk menerima *request* dan data dari *user* kemudian menentukan apa yang akan diproses oleh aplikasi.