

BAB II

STUDI PUSTAKA

2.1 Pengertian Sistem

Pengertian sistem menurut Erawati (2019), sistem adalah jaringan proses kerja yang saling berinteraksi satu sama lain guna untuk mencapai sebuah tujuan tertentu dan melakukan suatu kegiatan.

Sistem adalah kumpulan dari prosedur dan komponen yang saling berkaitan dan membentuk satu kesatuan untuk tujuan tertentu (Vanesa & Tasrif, 2022).

2.2 Pengertian Sistem Informasi

Menurut Satzinger (2017), menyatakan bahwa: Sistem informasi merupakan kumpulan dari komponen - komponen saling terkait dan bekerja sama untuk mengumpulkan, memproses, menyimpan, dan menyajikan setiap informasi yang dibutuhkan dalam proses bisnis dan aplikasi. Sistem ini dapat melibatkan penggunaan perangkat lunak, database, dan bahkan proses manual yang terkait.

Dengan pengertian sistem informasi diatas kesimpulan yang dapat diambil adalah sistem informasi merupakan sebuah sistem komputersasi untuk mengolah data. Sistem informasi ini mampu mengumpulkan, memproses dan menyimpan data. Tujuan dari sistem informasi ini adalah menyediakan informasi yang sesuai, tepat dan akurat kepada pengguna yang membutuhkan.

2.3 Magang

Magang atau *internship* merupakan suatu kegiatan yang bertujuan untuk menerapkan ilmu yang telah dipelajari dan mempraktekan kemampuan diri untuk mendapatkan pengalaman kerja yang baik sebagai persiapan untuk masa depan (Yulianto & Firdaus, 2021).

Magang adalah penempatan kerja dimana seorang mahasiswa bekerja untuk jangka waktu tertentu di suatu organisasi untuk mendapatkan pengalaman kerja di bidang yang diminati atau sesuai

dengan studinya yang nantinya dapat berguna untuk masa depan (Hotcours Indonesia 2022). Magang juga merupakan kesempatan untuk mahasiswa mengaplikasikan pengetahuan teoritis yang diperoleh dari pendidikan formal ke dalam konteks dunia kerja yang sebenarnya.

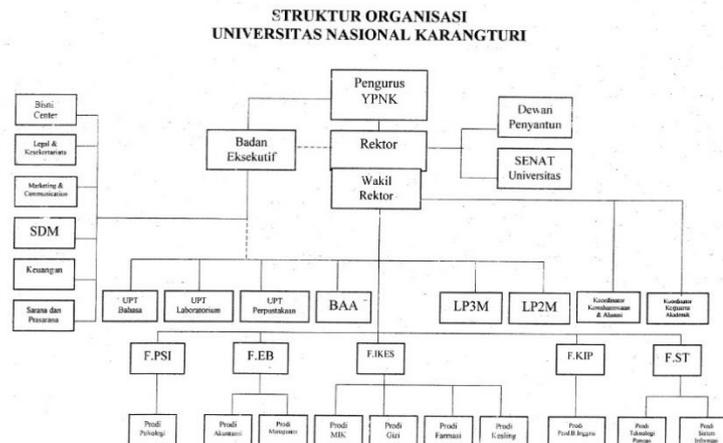
Selama magang, peserta magang juga akan terlibat dalam tugas-tugas, proyek, serta aktivitas sehari-hari yang berkaitan dengan bidang kerja yang dipilih. Tujuan magang adalah untuk memperoleh pemahaman yang lebih mendalam tentang profesi tertentu, mengembangkan keterampilan, membangun jaringan profesional, dan mendapatkan pengalaman kerja yang berharga

2.4 Universitas Nasional Karangturi

Universitas Nasional Karangturi adalah salah satu universitas swasta di Semarang yang mengusung nilai nasionalisme. Universitas ini berlokasi di jalan Raden Patah No.182-192 Rejomulyo, Semarang Timur Kota Semarang - Jawa Tengah. Universitas Nasional Karangturi berdiri sejak tahun 2017 yang didirikan oleh Yayasan Pendidikan Nasional Karangturi.

Universitas Nasional Karangturi memiliki dua fakultas yaitu yang pertama fakultas Sains dan Teknologi yang didalamnya terdapat tiga program studi yakni Sistem Informasi, Teknologi Pangan dan Manajemen Informasi Kesehatan. Yang kedua fakultas Ekonomi, Bisnis dan Humaniora yang menaungi empat program studi yakni Manajemen, Akuntansi, Pendidikan Bahasa Inggris dan Psikologi.

Unkartur menghadirkan pendekatan pembelajaran menggunakan metode terbaru guna memberikan pengalaman belajar selama masa studi tidak hanya di dalam ruang kelas, melainkan juga melalui pengalaman di luar kelas. Mahasiswa didorong untuk meraih pengalaman berharga selama berada di lingkungan kampus, sehingga mereka dapat tumbuh dalam kepribadian dan menggali potensi diri di dalam komunitas serta masyarakat sesuai dengan minat dan panggilan hati yang sejalan (unkartur.ac.id).



Gambar 2. 1 Struktur Organisasi (Sumber:Unkartur)

2.5 Merdeka Belajar Kampus Merdeka

Kebijakan Merdeka Belajar Kampus Merdeka (MBKM) menjadi salah satu kajian kebijakan yang tengah digalakan pemerintah untuk diterapkan pada dunia pendidikan perguruan tinggi. MBKM adalah mode pembelajaran pendidikan tinggi yang mandiri dan serbaguna yang dirancang untuk menciptakan komunitas pembelajaran kreatif yang tidak membatasi yang memenuhi kebutuhan mahasiswa (Darajatun & Ramdhany, 2021).

Tujuan MBKM adalah untuk meningkatkan kompetensi lulusan, baik *soft skills* maupun *hard skills* agar lebih siap dengan kebutuhan zaman, menyiapkan lulusan sebagai pemimpin masa depan bangsa yang unggul dan berkepribadian. Kebijakan MBKM ini sejalan dengan Peraturan Menteri Pendidikan dan Kebudayaan (Permendikbud) No. 3 Tahun 2020 tentang Standar Nasional PendidikanTinggi.

Berdasarkan Permendikbud No 3 Tahun 2020 pasal 15 ayat 1, bentuk kegiatan pembelajaran yang dapat dilakukan didalam prodi dan diluar prodi meliputi 8 program.

Berikut adalah kegiatan yang ada pada program Kampus Merdeka yaitu:



Gambar 2. 2 Program Kampus Merdeka
(Sumber: Kemdikbud.go.id)

1. Pertukaran Pelajar

Program pertukaran pelajar dilakukan antar perguruan tinggi dengan sistem kredit.

2. Magang/Praktik Kerja

Program magang dibuat karena kurangnya pengalaman kerja para lulusan perguruan tinggi sehingga mereka kurang siap saat dihadapi pada dunia kerja.

3. Asistensi mengajar di satuan pendidikan

Program ini dibuat karena kualitas pendidikan disekolah dasar dan menengah masih sangat rendah (PISA 2018 peringkat indonesia berada pada posisi nomor 7 dari bawah). Program ini dibuat dengan tujuan (1) Memberikan kesempatan kepada mahasiswa untuk memperdalam praktek untuk menjado guru disekolah; (2) Membantu meningkatkan pemerataan kualitas pendidikan; (3) Membantu mengisi keterbatasan guru serta kurangnya kualitas tenaga pendidik didaerah yang membuntuhkan (Fuadi, 2020).

4. Penelitian/Riset

Program ini dibuat untuk mahasiswa yang mempunyai *passion* menjadi peneliti. Program ini diharapkan dapat membangun cara berpikir kritis sehingga mereka dapat mendalami dan mampu melakukan metode riset dengan baik.

5. Proyek kemanusiaan

Program ini dibuat karena melihat Indonesia banyak mengalami bencana alam. Program ini bertujuan menyiapkan mahasiswa yang menjunjung tinggi nilai kemanusiaan serta melatih mahasiswa memiliki kepekaan sosial dalam menyelami permasalahan dan ikut memberikan solusi sesuai dengan keahlian masing-masing.

6. Kegiatan wirausaha

Program ini dicanangkan untuk mendorong dan dapat mengembangkan minat mahasiswa dalam bidang wirausaha.

7. Studi/Proyek independen

Program ini dibuat bertujuan untuk membantu mahasiswa yang bisa membuat sebuah karya yang inovatif untuk dilombakan ditingkat nasional maupun internasional.

8. Membangun Desa/Kuliah Kerja Nyata Tematik.

Program ini dibuat untuk memberikan pengalaman belajar kepada mahasiswa untuk hidup ditengah masyarakat.

Melalui program-program *experiential learning* yang memiliki jalur yang fleksibel, diharapkan mahasiswa memiliki kesempatan untuk mengembangkan potensinya sesuai dengan *passion* dan bakat yang dimiliki (Widagdo et al., 2022). Program ini dirancang agar mahasiswa dapat terlibat dalam pengalaman nyata yang relevan dengan minat mereka, sehingga mereka dapat memperoleh pemahaman yang lebih mendalam dan keterampilan yang sesuai dengan bidang yang mereka ambil. Dengan jalur yang fleksibel, mahasiswa memiliki kebebasan untuk menyesuaikan pengalaman belajar mereka sesuai dengan tujuan dan keinginan mereka

sendiri, sehingga mereka dapat memaksimalkan pengembangan diri mereka secara individual.

2.6 Unified Modelling Language (UML)

Menurut Prihandoyo (2018), Pemodelan *Unified Modeling Language* (UML), pemodelan visual yang digunakan untuk merancang atau membuat suatu aplikasi perangkat lunak yang berorientasi pada objek.

Menurut Gushelmi (2017), UML merupakan alat perancangan sistem yang berorientasi pada objek (*object oriented*).

Berdasarkan teori diatas dapat disimpulkan bahwa UML adalah metode analisis sistem atau sarana perancangan sistem yang menjadi penerjemah antara pengembang dan pengguna. Seorang programmer dapat dengan mudah memahami, menganalisa dan mempermudah pembuatan suatu program menggunakan UML. Sebuah program aplikasi biasanya berupa sistem yang digunakan dan diimplementasikan dalam jangka waktu yang cukup panjang. Program aplikasi bukan hanya digunakan pada saat itu saja, melainkan terus berlanjut. Karena program aplikasi digunakan pada waktu yang panjang, maka perlu adanya suatu analisis perencanaan, perancangan, dan pemodelan yang baik dan jelas. UML memiliki fungsi untuk membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek.

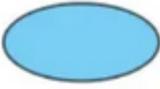
Penggunaan UML pada perancangan sistem informasi ini dikarenakan UML dapat mengidentifikasi kebutuhan sistem, merancang struktur dan hubungan antar komponen dan memodelkan aliran proses bisnis secara terstruktur dan terorganisir dengan merepresentasikan visual yang jelas menggunakan diagram yang mudah dipahami.

Jenis-jenis Diagram UML yaitu:

1. *Use Case Diagram*

Berfungsi untuk mendeskripsikan sebuah interaksi antara satu atau lebih *actor* dengan sistem informasi yang akan dibangun, dan dapat menggambarkan fungsi apa saja yang ada pada sebuah sistem informasi (Julianto & Setiawan, 2019).

Table 2. 1 *Simbol Use Case Diagram*

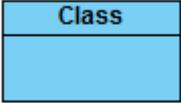
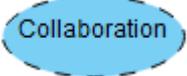
Simbol	Nama	Keterangan
	<i>Actor</i>	Entitas yang berinteraksi dengan sistem
	<i>Use case</i>	Aktivitas yang dapat dilakukan actor pada sistem
	<i>Association</i>	Hubungan antara actor dengan use case
	<i>System</i>	Sistem yang sedang dikembangkan
	<i>Include</i>	Satu use case termasuk bagian dari use case lain
	<i>extend</i>	Satu use case dapat diperluas dengan use case lain

	<i>Dependency</i>	Ketergantungan antara elemen-elemen diagram
	<i>Generalization</i>	Satu actor atau use case merupakan generalisasi dari yang lain.
	<i>Realization</i>	Implementasi dari satu use case oleh yang lain
	<i>Collaboration</i>	Dua atau lebih actor dan use case yang terhubung
	<i>Note</i>	Penjelasan tambahan terkait elemen-elemen diagram
	<i>Anchor</i>	Hubungan teks note dengan elemen-elemen diagram lain.

2. Class Diagram

Class diagram memberikan gambaran tentang bagaimana hubungan antara *class* dalam sistem yang sedang dikembangkan dan bagaimana *class-class* tersebut berkolaborasi untuk mencapai tujuan tertentu (Fu'ady & Suhendar, 2020). Diagram ini menunjukkan struktur dan hubungan antara kelas-kelas tersebut, termasuk relasi pewarisan, asosiasi dan ketergantungan antar kelas.

Table 2. 2 Simbol Class Diagram

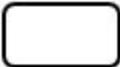
Simbol	Nama	Keterangan
	<i>Generalization</i>	Hubungan dimana objek anak(<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk(<i>ancestor</i>)
	<i>N-ary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi satu aktor
	<i>Realization</i>	Operasi yang benar benar dilakukan oleh suatu objek

	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung pada elemen yang tidak mandiri
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan yang lain

3. *Activity Diagram*

Menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem yang ada pada perangkat lunak bagaimana aktivitas dimulai, kejadian yang mungkin terjadi, serta bagaimana berakhirnya sistem tersebut (Pratama & Marjun, 2022). Perbedaan *use case diagram* dan *activity diagram* adalah untuk *use case* menggambarkan bagaimana *actor* berjalan atau *actor* ketika menggunakan sistem untuk melakukan aktivitas. Sedangkan *activity diagram* menggambarkan aktifitas sistem bukan apa yang dilakukan oleh aktor.

Table 2. 3 Simbol Activity Diagram

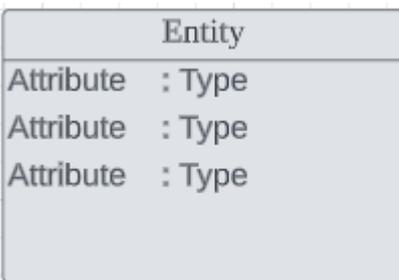
Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
	Percabangan/ <i>Decision</i>	Dimana ada pilihan aktivitas yang lebih dari satu
	Penggabungan/ <i>Join</i>	Dimana lebih dari satu aktivitas digabungkan jadi satu
	Status akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
	Swimlane	Memisahkan organisasi bisnis yang

		bertanggung jawab terhadap aktivitas yang terjadi
--	--	---------------------------------------------------

2.7 Entity Relationship Diagram (ERD)

Diagram Hubungan Entitas, yang juga dikenal sebagai Entity Relationship Diagram (ERD), adalah suatu bentuk model data yang menggunakan notasi grafis untuk menggambarkan struktur dan hubungan antara entitas dalam suatu sistem atau basis data (Wikipedia). Model ini digunakan dalam pemodelan data konseptual untuk memberikan gambaran visual tentang bagaimana entitas berinteraksi dan terhubung satu sama lain dalam konteks bisnis atau aplikasi tertentu. Dalam ERD, entitas mewakili objek yang memiliki informasi atau data yang relevan dalam sistem. Hubungan antara entitas diilustrasikan melalui garis dan simbol yang mencerminkan cara data saling terhubung dan berinteraksi. Diagram ini membantu pengembang dan analis dalam merancang struktur basis data dengan lebih jelas dan sistematis, serta mempermudah pemahaman tentang bagaimana informasi akan disimpan, dikelola, dan diakses. Berikut adalah simbol-simbol yang ada pada ERD:

Table 2. 4 Simbol ERD

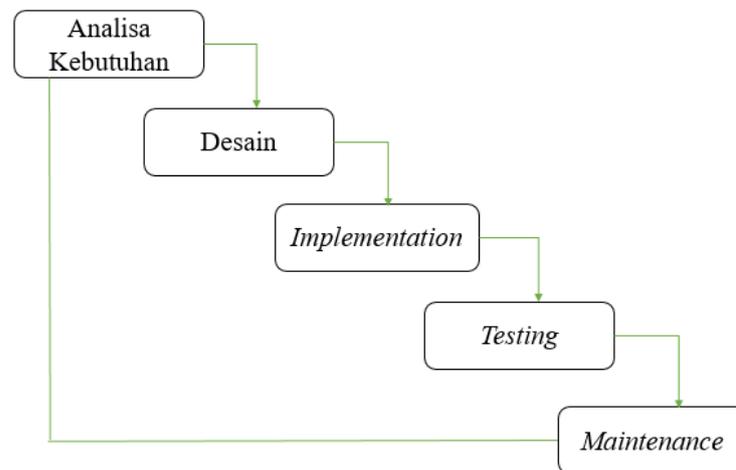
Simbol	Keterangan
	<p><i>Entity</i>: Merupakan sebuah objek, orang atau kejadian yang merupakan tempat data dikumpulkan.</p> <p><i>Attribute</i>: Atribut menggambarkan karakteristik</p>

	<p>atau informasi yang terkait dengan entitas.</p> <p><i>Type:</i> Adalah tipe data dari atribut.</p>
	<p><i>One to One</i> : Relasi "<i>one-to-one</i>" mengacu pada hubungan antara dua entitas di mana satu entitas dari setiap sisi relasi hanya dapat terhubung dengan satu entitas di sisi lainnya. Dengan kata lain, setiap entitas pada satu sisi relasi memiliki pasangan tunggal di sisi lainnya, dan sebaliknya.</p>
	<p><i>One to Many</i> : Mengindikasikan bahwa setiap entitas di sisi satu relasi hanya dapat terhubung dengan satu entitas di sisi lainnya, tetapi entitas di sisi lainnya dapat terhubung dengan banyak entitas di sisi satu relasi.</p>
	<p><i>One (and only one)</i> : dihubungkan dengan satu dan hanya satu</p>
	<p><i>Zero or One</i> : Mengindikasikan bahwa entitas di satu sisi relasi bisa memiliki nol entitas yang terkait atau satu entitas yang terkait di sisi lainnya.</p>

2.8 Metode *Waterfall*

Waterfall merupakan model klasik yang memiliki sifat berurut dalam merancang serta membangun sebuah sistem informasi, dimana sebuah langkah harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase berikutnya (Pratama & Marjun, 2022). Keuntungan menggunakan metode ini adalah metode ini dapat mengurangi kesalahan dan memungkinkan sedikit terjadinya perubahan yang dilakukan selama proyek berlangsung.

Berikut adalah tahapan yang dipakai dalam model *waterfall*, yaitu:



Gambar 2. 3 Metode *Waterfall*

1. Analisa Kebutuhan

Proses pengumpulan kebutuhan dilakukan untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami, perangkat lunak seperti apa yang dibutuhkan oleh user. Hasil analisa ini digunakan sebagai dasar untuk merancang sistem informasi magang MBKM. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasi, untuk perangkat lunak yang dibutuhkan yaitu:

- a. Bahasa pemrograman PHP
- b. *MySQL* sebagai *database*
- c. *Framework Codeigniter*

2. Desain

Setelah kebutuhan yang telah ditentukan pada tahap sebelumnya di analisis. Tahap berikutnya adalah mendesain arsitektur secara menyeluruh. Desain sistem akan membantu menentukan perangkat keras dan persyaratan sistem yang akan membantu dalam mendefinisikan arsitektur sistem (Pratama & Marjun, 2022).

3. *Implementation*

Tahap ini adalah sebagai penerjemahan desain sistem, programmer mulai melakukan pengkodean (coding) sesuai dengan desain yang sudah dibuat di fase sebelumnya. Programmer akan menerjemahkan perintah yang diminta oleh pengguna.

4. *Testing*

Pada tahap ini dilakukan pengujian pada perangkat lunak dengan tujuan memastikan bahwa semua komponen telah diuji untuk mengurangi kesalahan dan memastikan outputnya sesuai.

5. *Maintenance*

Maintenance perangkat lunak terjadi pada tahap ini. Dalam tahap pemeliharaan ini, proses pengembang dapat diulang dimulai dari analisis spesifikasi, untuk melakukan perubahan pada perangkat lunak yang sudah ada, namun tidak untuk membuat perangkat lunak yang baru (Hidayati, 2019).

2.9 Hypertext Preprocessor (PHP)

PHP adalah sebuah bahasa pemrograman berbasis *server-side* yang memiliki kemampuan untuk mengolah skrip PHP menjadi skrip web. Hal ini memungkinkan pengembang untuk menciptakan tampilan website yang menarik dari sisi *client* (Fahlevi et al., 2021). Dengan PHP, skrip-skip yang ditulis dapat diproses oleh server dan diubah menjadi halaman web yang dinamis, interaktif, dan responsif. Dengan demikian, PHP memberikan fleksibilitas dan kemampuan yang diperlukan untuk menghasilkan pengalaman pengguna yang memikat melalui halaman web.

Bahasa pemrograman PHP digunakan pada pembuatan sistem ini dikarenakan PHP memiliki sintaks yang mudah dipahami dan dipelajari, selain itu kepopularitasan bahasa pemrograman ini membuat banyak sumber daya, tutorial dan *library* yang tersedia untuk membantu dalam pengembangan sistem informasi.

Berikut beberapa keunggulan yang dimiliki oleh bahasa pemrograman PHP:

1. PHP diterbitkan secara gratis.
2. Beberapa server seperti *Apache*, *Microsoft IIS*, *PWS*, *AOLServer*, *phhttpd*, *fhhttpd*, dan *Xitami* mampu menjalankan PHP.
3. Tingkat akses PHP lebih cepat serta memiliki tingkat keamanan yang tinggi.
4. Beberapa *database* yang sudah ada, baik yang bersifat gratis ataupun komersial sangat mendukung akses PHP, diantaranya *MySQL*, *PostgreSQL*, *mSQL*, *Informix*, dan *Microsoft SQL server*.
5. PHP mampu berjalan di Linux sebagai *platform* sistem operasi utama bagi PHP, tetapi dapat juga berjalan di *FreeBSD*, *Unix*, *Solaris*, *Windows*, dan yang lainnya.

2.10 Database MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen *database*. *Database* ini dirancang untuk menyimpan, mengelola dan mengakses data dalam suatu sistem.

MySQL memiliki fitur yang kuat dan fleksibel serta mendukung bahasa *SQL* (*Structured Query Language*) untuk melakukan pengolahan data (Wikipedia 2023).

Keunggulan *MySQL* antara lain adalah:

1. Kecepatan akses yang tinggi
2. Dukungan untuk multi-user
3. Kemampuan untuk mengelola jumlah data yang besar

4. Kompatibilitas dengan berbagai bahasa pemrograman. *MySQL* dapat diintegrasikan dengan mudah dengan bahasa pemrograman populer seperti *PHP*, *Java*, dan *Python*, serta banyak kerangka kerja web yang ada saat ini.

2.11 Codeigniter

Codeigniter adalah suatu *framework PHP* yang menggunakan metode *Model View Controller (MVC)*, yang memungkinkan programmer untuk membangun aplikasi atau situs web dengan lebih mudah tanpa harus membangun dari awal (Sallaby & Kanedi, 2020).

Codeigniter menjadi salah satu *framework* yang populer dalam pengembangan aplikasi web karena *codeigniter* sangat mudah dipahami dan fleksibel. Selain itu juga *framework codeigniter* memiliki dokumentasi yang lengkap sehingga memudahkan programmer mempelajari *codeigniter* dengan cepat.

MVC memungkinkan programmer bekerja secara terarah. Dengan adanya konsep ini programmer dapat dengan mudah melakukan perubahan pada satu komponen tanpa mempengaruhi yang lain.

Penggunaan *framework codeigniter* dalam pembuatan sistem informasi ini dikarenakan *codeigniter* memiliki pengaturan yang sudah ditentukan secara default, sehingga mengurangi kesenjangan dalam pembelajaran dan mempercepat proses pengembangan selain itu *codeigniter* juga memiliki dokumentasi yang *komprehensif* dan mudah diakses. Dokumentasi ini menjelaskan secara rinci tentang fitur dan konsep *framework*, serta menyediakan contoh *code* yang berguna.

Berikut adalah beberapa kelebihan dari *framework codeigniter* jika dibandingkan dengan *framework* lainnya:

1. *Open-Source* : *Codeigniter* merupakan kerangka kerja dengan lisensi *Apache/BSD open-source*, yang berarti dapat digunakan secara bebas dan gratis.

2. Ukuran yang Kecil : Ukuran *Codeigniter* yang relatif kecil dibandingkan dengan *framework* lain yang lebih besar. Hal ini mengakibatkan kebutuhan *resource* yang lebih sedikit baik dalam eksekusi maupun penyimpanan.
3. Dokumentasi yang lengkap: setiap paket instalasi *CodeIgniter* sudah disertai panduan untuk pengguna yang sangat bagus dan lengkap untuk dijadikan permulaan, bahasanya pun mudah dimengerti.
4. Menggunakan Konsep MVC : *Codeigniter* menerapkan konsep MVC yang memungkinkan pemisahan antara logika aplikasi dan tampilan.

Berikut adalah penjelasan dari konsep MVC :

a. *Model*

Merupakan bagian yang bertanggung jawab untuk mengolah data dan berinteraksi dengan database. Seperti mengambil data dan melakukan pengolahan database lainnya yang diperlukan oleh aplikasi (Sulistyo & Widodo, 2021).

b. *View*

Merupakan bagian yang bertanggung jawab menampilkan informasi kepada pengguna. Semua yang berhubungan dengan tampilan dikumpulkan di *view*.

c. *Controller*

Controller berfungsi sebagai penghubung antara *model* dan *view*. *Controller* merupakan pengontrol yang menangani permintaan dari pengguna dan mengintruksikan *model* dan *view* melakukan tugasnya.

2.12 Penelitian Terkait

Table 2. 5 Penelitian Terkait

No	Nama dan judul	Masalah	Solusi	Hasil
1	M Satria Putra Utama, Denny Sagita Rusdianto, Faizatul Amalia (2022) Rancang Bangun Sistem Informasi Praktik Kerja Lapangan berbasis Web (Studi Kasus: Program Studi Teknik Informatika Universitas Brawijaya)	Mahasiswa yang ingin melakukan PKL sering mengalami kesulitan untuk menentukan lokasi tempat melaksanakan PKL.	Pengembangan sistem ini menggunakan metode waterfall model. Sistem ini mengimplementasikan web dengan framework Codeigniter Fitur utama pada sistem ini yaitu menambah data informasi, menambah data instansi, menyetujui penempatan mahasiswa, melihat nilai praktik dan monitoring PKL mahasiswa..	Hasil yang diperoleh 100% valid pada pengujian fungsional dan pengujian non fungsional berupa pengujian usability,berdasarkan interpretasi SUS skor maka sistem dikategorikan sudah sesuai dengan kebutuhan pengguna.
2	Anri Qasthari Adyan , Boko Susilo, Desi Andreswari (2020) Penempatan Praktik Kerja Lapangan Berdasarkan Nilai	siswa mendapatkan lokasi tempat praktek kerja lapangan (PKL) sesuai dengan keahlian dan kemampuan siswa tersebut.	dapat membantu dengan cara membangun sebuah sistem pendukung keputusan penempatan praktik kerja lapangan siswa secara lebih baik menggunakan metode pembobotan ROC (Rank	Hasil dari penelitian adalah sistem yang digunakan oleh pengguna untuk memberikan rekomendasi tempat praktik kerja lapangan bagi siswa yang sesuai dengan kriteria yang ditentukan dengan

	Kompetensi Dasar Dan Nilai Sikap Siswa Menggunakan Metode Pembobotan Rank Order Centroid Dan Metode Profile Matching (Studi Kasus : SMKN 1 Kota Bengkulu)		Order Centroid) dan metode Profile Matching. Diharapkan dapat membantu staf ataupun guru pada bidang Humas yang mengurus penempatan siswa kelas XI dalam melakukan Praktik Kerja Lapangan (PKL) di SMKN 1 Kota Bengkulu.	menggunakan metode pembobotan Rank Order Centroid dan Profile Matching.
3	I Gusti Ayu Made Widyari, I Gusti Lanang Agung Raditya Putra, I Gede Putu Krisna Juliharta "Rancang Bangun Sistem Informasi Data Siswa Praktik Kerja Lapangan (PKL) Berbasis Web Responsive Pada SMK	Dalam kegiatan PKL ini, pihak sekolah SMK TI Udayana masih menggunakan sistem manual dalam mendata siswa yang mengakibatkan kinerja pihak sekolah menjadi kurang efektif.	Pengimplementasian sistem PKL bertujuan agar meningkatkan kinerja pihak sekolah dalam kegiatan praktek kerja lapangan dan memudahkan siswa dalam mencari informasi jadwal praktek kerja lapangan. Penelitian ini dilakukan dengan memanfaatkan web responsive yang dapat memudahkan user pada saat menggunakan	Sistem ini dapat digunakan oleh empat user yaitu operator, kepala sekolah, guru pembimbing dan siswa. Penyebaran kuisisioner kepada responden atau pengguna sistem menghasilkan sistem informasi data siswa PKL yang efektif dikembangkan untuk menjangkakan kinerja pihak sekolah SMK TI Udayana.

	TI Udayana"		sistem informasi PKL.	
4	Nurlela, Nur Ilman, Nur Laela Sayuti "Perancang an Sistem Informasi Praktek Kerja Lapangan Berbasis Web Pada Amik Ibnu Khaldun Palopo "	Dalam melaksanakan praktek kerja lapangan di AMIK Ibnu Khaldun Palopo pengolahan data belum sepenuhnya terkomputerisasi, seperti pada proses pengajuan praktek kerja lapangan masih terdapat kesulitan dalam pengolahan data dan kurangnya informasi dalam menentukan pembimbing dan tempat praktek kerja lapangan.	Membuat sistem informasi praktek kerja lapangan (PHP) berbasis web yang dapat mempermudah dalam pengolahan data praktek kerja lapangan.	Dari hasil pengujian yang telah didapatkan dapat disimpulkan bahwa perangkat lunak bebas dari kesalahan sintaks dan secara fungsional dapat mengeluarkan hasil yang sesuai dengan yang diharapkan sehingga aplikasi ini layak untuk dipergunakan pada AMIK Ibnu Khaldun Palopo. Dengan adanya sistem informasi praktek kerja lapangan pada AMIK Ibnu Khaldun Palopo diharapkan dapat membantu dan memudahkan pihak kampus dalam mengelola informasi pengajuan praktek kerja lapangan.
5	Didik Indrayana, Maria Ulfah, Somantri "Rancang Bangun Aplikasi Pencarian	Sulitnya mencari perusahaan atau instansi yang dapat menerima mahasiswa secara linear sangat	Mempermudah mahasiswa mencari informasi mengenai tempat magang dengan mudah dan sederhana sesuai jurusan yang	Mahasiswa dapat memilih perusahaan yang di inginkan sesuai dengan jurusannya. Mahasiswa dapat menerima informasi

	Tempat Magang Untuk Mahasiswa Berbasis Android"	menjadi permasalahan pada kegiatan magang mahasiswa.	diharapkan serta dapat dioperasikan dimanapun user berada menggunakan piranti mobile device berbasis Android.	mengenai data-data perusahaan serta rute perjalanan menuju perusahaan yang akan dituju. Harus memiliki jaringan internet juga mengaktifkan GPS untuk dapat mengoperasikan aplikasi "Sistem Informasi Geografis (SIG) pencarian tempat magang ini. Perancangan dan implementasi dari SIG menghasilkan beberapa fungsi yang dapat digunakan oleh mahasiswa yaitu fungsi untuk mencari lokasi jalan atau suatu tempat dan fungsi untuk melihat informasi detail dari hasil pencarian tersebut.
--	-------------------------------------------------	------------------------------------------------------	---------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------