

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Penelitian serupa pernah dilakukan oleh Gede Jaya Widhi Aryadi (2020) yang berjudul “Pembangunan Sistem Informasi Praktik Kerja Industri (Prakerin) berbasis Website menggunakan Model Pengembangan Perangkat Lunak Waterfall Studi Kasus : SMK Plus Almaarif Singosari”. Sistem yang dibuat bertujuan untuk memudahkan peserta didik dalam mencari informasi industri yang sesuai dengan kemampuan atau kompetensinya tanpa mengalami kendala jarak dan waktu serta dapat menjadi sarana penyajian informasi yang akurat, cepat, lengkap dan dapat membantu siswa dan orang tua/wali dalam memahami tentang pelaksanaan prakerin/pkl. Sistem ini dibangun menggunakan bahasa pemrograman PHP, framework Laravel dan database MySQL[6].

Penelitian lainnya yang serupa pernah dilakukan oleh Mochamad Baiduri (2019) yang berjudul “Perancangan Sistem Informasi Praktek Kerja Industri di SMK Bakti Nusantara 666 Cileunyi”. Sistem yang dibuat bertujuan untuk mempermudah pelaksanaan prakerin/pkl di SMK Bakti Nusantara 666 Cileunyi terutama dalam pendataan siswa prakerin/pkl, pengelolaan agenda harian siswa, penentuan pembimbing prakerin/pkl, dan penilaian prakerin/pkl. Metode pengembangan sistem yang digunakan adalah metode waterfall. Sistem ini dibangun menggunakan bahasa pemrograman PHP dan database MySQL[7].

Penelitian lainnya yang serupa pernah dilakukan oleh Riska El Fauzi (2021) yang berjudul “Rancang Bangun Sistem Informasi Praktek Kerja Industri (Prakerin) Berbasis Web”. Sistem yang dibuat bertujuan untuk memudahkan siswa mengakses informasi yang diberikan oleh panitia prakerin, melakukan pendaftaran prakerin, melaksanakan prakerin sesuai tempat yang dipilih/ditentukan. Metode pengembangan sistem yang digunakan adalah metode waterfall. Sistem ini dibangun menggunakan bahasa pemrograman PHP, framework Laravel dan database MySQL[8].

Penelitian lainnya yang serupa pernah dilakukan oleh Meiftahudin Triyas Subarkah (2020) yang berjudul “Sistem Informasi Pengelolaan Praktik Kerja Industri (Studi Kasus: Smk Ma’arif 1 Kroya)”. Sistem yang

dibuat bertujuan untuk mempermudah proses pengajuan, pembuatan surat, dan bimbingan, mempermudah dan mengefektifkan pengolahan data sehingga kegiatan prakerin dapat berjalan lebih baik. Metode pengembangan sistem yang digunakan adalah metode waterfall. Sistem ini dibangun menggunakan bahasa pemrograman PHP, dan database MySQL[9].

Penelitian lainnya yang serupa pernah dilakukan oleh Muhamad Malik Mutoffar (2019) yang berjudul “Aplikasi Praktek Kerja Lapangan Berbasis Web Smkn 1 Majalaya”. Sistem yang dibuat bertujuan untuk memudahkan guru pembimbing sekolah dalam hal sistem monitoring siswa, administrasi, berkas ujian laporan dan penilaian sidang prakerin serta dapat menjadi solusi serta mempermudah pelaksanaan praktik kerja lapangan. Metode pengembangan sistem yang digunakan adalah metode waterfall. Sistem ini dibangun menggunakan bahasa pemrograman PHP, dan database MySQL[10].

Tabel 2.1 Penelitian Sebelumnya

1	Judul	Pembangunan Sistem Informasi Praktik Kerja Industri (Prakerin) berbasis Website menggunakan Model Pengembangan Perangkat Lunak Waterfall Studi Kasus : SMK Plus Almaarif Singosari
	Peneliti (tahun)	Gede Jaya Widhi Aryadi, Ahmad Afif Supianto dan Widhy Hayuhardhika Nugraha Putra (2020)
	Tujuan	Memudahkan peserta didik dalam mencari informasi industri yang sesuai dengan kemampuan atau kompetensinya tanpa mengalami kendala jarak dan waktu serta dapat mejadi sarana penyajian informasi yang akurat, cepat, lengkap dan dapat membantu siswa dan orang tua/wali dalam memahami tentang pelaksanaan prakerin/pkl.
	Metode	<i>Waterfall</i>
	Hasil	Penelitian ini menghasilkan sistem informasi yang diberi nama “SIPRAKERIN”. Terdapat 4 pengguna yang akan menjalankan sistem, yaitu Humas, Kepala Prodi, Guru Pembimbing dan Siswa.

2	Judul	Perancangan Sistem Informasi Praktek Kerja Industri di SMK Bakti Nusantara 666 Cileunyi
	Peneliti (tahun)	Mochamad Baiduri dan Erna Susilawati (2019)
	Tujuan	Mempermudah pelaksanaan prakerin/pkl di SMK Bakti Nusantara 666 Cileunyi terutama dalam pendataan siswa prakerin/pkl, pengelolaan agenda harian siswa, penentuan pembimbing prakerin/pkl, dan penilaian prakerin/pkl.
	Metode	<i>Waterfall</i>
	Hasil	Penelitian ini menghasilkan sistem informasi praktik kerja industri di SMK Bakti Nusantara 666 Cileunyi dan telah dapat membantu siswa dan guru hubin dalam pendaftaran dan monitoring prakerin dan dapat membantu guru penguji dan guru hubin dalam mengelola nilai akhir prakerin.
3	Judul	Rancang Bangun Sistem Informasi Praktek Kerja Industri (Prakerin) Berbasis Web
	Peneliti (tahun)	Riska El Fauzi dan Mohammad Adri (2021)
	Tujuan	Memudahkan siswa mengakses informasi yang diberikan oleh panitia prakerin, melakukan pendaftaran prakerin, melaksanakan prakerin sesuai tempat yang dipilih/ditentukan.
	Metode	<i>Waterfall</i>
	Hasil	Penelitian ini menghasilkan sistem informasi pengelolaan praktik kerja industri dengan 4 pengguna, yaitu Admin, Siswa, Guru pembimbing dan Pihak DU/DI yang dapat digunakan sebagai media alternative dalam pengelolaan praktik kerja industri di SMK Negeri 2 Padang Panjang.

4	Judul	Sistem Informasi Pengelolaan Praktik Kerja Industri (Studi Kasus: Smk Ma'arif 1 Kroya)
	Peneliti (tahun)	Meifatahudin Triyas Subarkah, Dwi Krisbiantoro dan Abdul Jahir (2020)
	Tujuan	Memper memudahkan proses pengajuan, pembuatan surat, dan bimbingan, mempermudah dan mengefektifkan pengolahan data sehingga kegiatan prakerin dapat berjalan lebih baik.
	Metode	<i>Waterfall</i>
	Hasil	Penelitian ini menghasilkan sistem informasi pengelolaan prakerin di SMK Ma'arif 1 Kroya dengan 3 pengguna yaitu administrator, siswa sebagai peserta prakerin, dan guru pembimbing yang dapat memudahkan POKJA Prakerin.
5	Judul	Aplikasi Praktek Kerja Lapangan Berbasis Web SMKN 1 Majalaya
	Peneliti (tahun)	Muhamad Malik Mutoffar, Imam Yunianto dan Heri Afitriansyah (2019)
	Tujuan	Membuat proses administrasi menjadi lebih terkomputerisasi, memudahkan guru pembimbing sekolah dalam hal sistem monitoring siswa, administrasi, berkas ujian laporan dan penilaian sidang prakerin serta dapat menjadi solusi serta mempermudah pelaksanaan praktik kerja lapangan.
	Metode	<i>Waterfall</i>
	Hasil	Penelitian ini menghasilkan sistem informasi praktik kerja lapangan di SMKN 1 Majalaya, sehingga pelaksanaan pkl menjadi terstruktur ,pengelolaan data menjadi lebih efektif dan efisien serta dapat mempermudah pelaksanaan praktik kerja lapangan di SMKN 1 Majalaya.

Dilihat dari penelitian- penelitian yang terdapat pada table 2.1, terdapat persamaan dengan penelitian “Sistem Informasi Praktik Kerja Lapangan di SMK Negeri 1 Cilacap Berbasis Website” yang diajukan, yaitu sistem berbasis website, menggunakan bahasa pemrograman PHP. Terdapat juga perbedaannya yaitu dalam penelitian “Sistem Informasi Praktik Kerja Lapangan di SMK Negeri 1 Cilacap Berbasis Website” peneliti menggunakan metode *prototyping* sebagai metode pengembangan sistem yang digunakan, serta menggunakan *PostgreSQL* sebagai *database*. Perbedaan lainnya adalah siswa dapat mengunduh kartu peserta pkl dan dapat melakukan penilaian terhadap tempat praktik kerja lapangan (PKL) sebagai acuan apakah rekomendasi atau tidak untuk pelaksanaan praktik kerja lapangan (PKL) pada tahun berikutnya, pembimbing sekolah dapat memasukkan hasil monitorig ke sistem, terdapat pembimbing industri yang dapat melakukan penilaian siswa pkl dan menggunggah sertifikat pkl, dan siswa dapat mengunduh sertifikat pkl.

2.2 Landasan Teori

2.2.1 Sistem Informasi

Sistem informasi adalah pengaturan orang, data, proses, dan teknologi informasi yang berinteraksi untuk mengumpulkan, memproses, menyimpan, dan menyediakan sebagai output informasi yang diperlukan untuk mendukung organisasi atau instansi[11].

2.2.2 Framework

Framework adalah komponen pemrograman yang siap *re-use* (bisa digunakan ulang) kapan saja, sehingga *programmer* tidak harus membuat skrip yang sama untuk tugas yang sama. Misalkan *programmer* ingin halaman-halaman web menampilkan data dengan paginasi (*paging*) halaman, *framework* telah menyediakan fungsi *paging* tersebut sedangkan *programmer* cukup menggunakan fungsi tersebut pada saat *coding*, tetapi tentu dengan kaidah- kaidah yang ditetapkan oleh masing - masing *framework*[12].

2.2.3 Laravel

Laravel adalah sebuah *framework* web berbasis PHP yang *open-source* dan tidak berbayar, diciptakan oleh Taylor Otwell dan diperuntukkan untuk pengembangan aplikasi web yang menggunakan pola MVC. Struktur pola MVC pada *laravel* sedikit berbeda pada struktur

pola MVC pada umumnya. Di *laravel* terdapat *routing* yang menjembatani antara *request* dari *user* dan *controller*. Jadi *controller* tidak langsung menerima *request* tersebut[12].



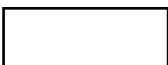

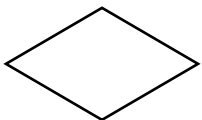
2.2.4 UML (*Unified Modelling Language*)

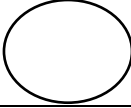
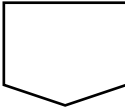


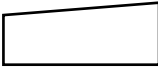
UML adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek[14].

a. *Flowchart*

Flowchart merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempengaruhi penyelesaian masalah yang khususnya perlu dipelajari dan dievaluasi lebih lanjut[15]. Simbol *flowchart* dapat dilihat pada tabel 2.2.

Tabel 2.2 Simbol *Flowchart*


No.	Nama	Simbol	Keterangan
1.	Terminal		Menyatakan permulaan atau akhir suatu program
2.	<i>Input dan Output</i>		Menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya
3.	Proses		Menyatakan proses yang dilakukan oleh komputer
4.	Simbol arus/ <i>flow</i>		Menyatakan jalannya arus suatu proses
5.	<i>Decision / keputusan</i>		Menunjukkan suatu kondisi tertentu yang akan menghasilkan data kemungkinan dengan jawaban ya atau tidak






No.	Nama	Simbol	Keterangan
6.	<i>Connector (on-page)</i>		Menghubungkan proses ke proses lain dalam halaman yang sama
7.	<i>Connector (off-page)</i>		Menghubungkan proses ke proses lain dalam halaman yang berbeda
8.	<i>Manual Operation</i>		Menunjukkan kegiatan yang tidak dilakukan oleh komputer
9.	Dokumen		Mencetak <i>output</i> dalam dokumen melalui printer
10.	<i>Manual Input</i>		Memasukkan data secara manual dengan menggunakan online keyboard

b. Use case diagram

Use case diagram adalah diagram yang bekerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan suatu sistem tersendiri melalui sebuah cerita bagaimana sebuah sistem dipakai[16]. Simbol *use case diagram* dapat dilihat pada tabel 2.3.

Tabel 2.3 Simbol *Use Case Diagram*


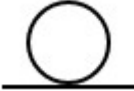
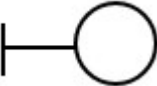



No.	Nama	Simbol	Keterangan
1.	<i>Use Case</i>		Fungsi yang disediakan sistem untuk unit yang saling bertukar pesan antar unit dan aktor

No.	Nama	Simbol	Keterangan
2.	<i>Actor</i>		Orang yang berinteraksi dengan sistem informasi yang dibuat
3.	<i>Asosiasi</i>		Penghubung komunikasi antar aktor dan <i>use case</i>
4.	<i>Ekstend</i>		Penghubung komunikasi antar aktor dan <i>use case</i>
5.	<i>Include</i>		Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya.
6.	Generalisasi		Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i> .

c. *Sequence Diagram*

Sequence Diagram merupakan diagram yang digunakan untuk menggambarkan interaksi antar objek di dalam dan di sekitar sistem yang berupa message yang digambarkan terhadap waktu[16]. Simbol *Sequence Diagram* dapat dilihat pada tabel 2.4.





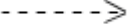

Tabel 2.4 Simbol *Sequence Diagram*

No.	Nama	Simbol	Keterangan
1.	<i>Actor</i>		Menggambarkan orang yang berinteraksi dengan sistem
2.	<i>Entity Class</i>		Menggambarkan hubungan yang akan dilakukan
3.	<i>Boundary Class</i>		Menggambarkan sebuah gambaran dari <i>form</i>
4.	<i>Control Class</i>		Menggambarkan hubungan antara <i>boundary</i> dengan tabel
5.	<i>Lifeline</i>		Menggambarkan tempat mulai dan berakhirnya <i>message</i>
6.	Line Message		Menggambarkan pengiriman <i>message</i> antar <i>class</i>

d. *Class Diagram*

Class diagram merupakan diagram yang digunakan untuk melakukan visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak digunakan[16]. Simbol *Class Diagram* dapat dilihat pada tabel 2.5.

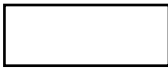
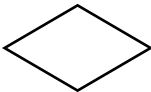


Tabel 2.5 Simbol *Class Diagram*

No.	Nama	Simbol	Keterangan
1.	<i>Generalization</i>		Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus)
2.	<i>Nary Association</i>		Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3.	<i>Class</i>		Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
4.	<i>Realization</i>		Operasi yang dilakukan oleh suatu objek
5.	<i>Dependency</i>		Relasi antarkelas dengan makna kebergantungan antarkelas
6.	<i>Association</i>		Diagram yang menghubungkan antara objek satu dengan objek lainnya

e. *Entity Relationship Diagram*

Entity Relationship Diagram adalah model teknik pendekatan yang menyatakan atau menggambarkan hubungan suatu model. Didalam hubungan ini tersebut dinyatakan yang utama dari ERD adalah menunjukan objek data (*Entity*) dan hubungan (*Relationship*), yang ada pada *Entity*[17]. Simbol *Entity Relationship Diagram* dapat dilihat pada tabel 2.6.

Tabel 2.6 Simbol *Entity Relationship Diagram*

No.	Nama	Simbol	Keterangan
1.	Entitas		Objek yang dapat diidentifikasi dalam lingkungan pemakai
2.	Relasi		Hubungan di antara sejumlah entitas yang berbeda
3.	Atribut		Mendeskripsikan karakter entitas
4.	Garis/ <i>Line</i>		Penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut

2.2.5 Basis Data (*Database*)

Basis Data atau *Database* adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut[18].

a. **DBMS (*Database Management System*)**

Database Management System (DBMS) adalah perangkat lunak yang memungkinkan pemakai untuk dapat menjelaskan, mengelola, dan mengontrol akses ke basis data. Komponen DBMS diklasifikasikan menjadi *Hardware dan Software, People dan Procedure, Data* [19].

b. ***PostgreSQL***

PostgreSQL merupakan *database* bersifat *open source* disertai kemampuannya yang bagus dibanding basis data lainnya. Adapun Fitur pada basis data *PostgreSQL* sangat lengkap, serta dapat mendukung aplikasi *database* dalam skala menengah maupun skala besar, serta memiliki fitur untuk replikasi *database*. Fitur- fitur yang disediakan antara lain *DB Mirror, PGPool, Slony, PGCluster*, dan lain-lain[20].

2.2.6 Praktik Kerja Lapangan (PKL)

Praktik Kerja Lapangan (PKL) merupakan pembelajaran bagi peserta didik SMK/MAK, SMALB, dan LKP yang dilaksanakan melalui praktik kerja di dunia kerja dalam jangka waktu tertentu sesuai dengan kurikulum dan kebutuhan dunia kerja[2].

2.2.7 Website Usability

Website usability merupakan suatu indikator keberhasilan sebuah *website* berinteraksi dengan pengguna dalam melaksanakan tugas tertentu dengan mudah. Ukuran keberhasilan dari *website usability* dilihat dari seberapa baik sebuah *website* dalam memberikan kualitas layanan kepada pengguna, mengurangi kemungkinan kesalahan pada sistem, serta memudahkan proses pembelajaran dan penggunaan. *Website usability* dapat mengetahui kualitas suatu *website*[21].

2.2.8 Pengujian Usability

Pengujian *usability* merupakan pengujian yang digunakan untuk mengukur kualitas pengalaman pengguna ketika berinteraksi dengan sistem atau perangkat lunak[22].

2.2.9 Metode Pengembangan Sistem

Metode *Prototyping* merupakan pengembangan sistem yang digunakan dalam pengembangan sistem ini. *Prototyping* merupakan proses yang digunakan untuk membantu pengembangan perangkat lunak dalam membentuk model perangkat lunak yang harus dibuat. Sistem dengan model *prototyping* mengizinkan pengguna agar mengetahui seperti apa tahapan sistem dibuat sehingga sistem mampu beroperasi dengan baik. Rancangan aplikasi awal mulanya berbentuk *mockup* selanjutnya akan dievaluasi oleh pengguna. Setelah *mockup* dievaluasi pengguna, tahap selanjutnya *mockup* menjadi bahan rujukan bagi pengembang *software* untuk merancang aplikasi[3].

Tahapan-tahapan dalam metode *Prototyping* adalah sebagai berikut[4]:

a. Tahap Pengumpulan Kebutuhan

Pada tahapan ini, pengembang dan pelanggan mendefinisikan format dan kebutuhan perangkat lunak secara keseluruhan,

mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.

- b. Tahap Membangun *Prototyping*
Pada tahapan ini, pengembang membuat desain sementara yang berpusat pada penyajian kepada pelanggan, misalnya; dengan membuat contoh input dan output.
- c. Tahap Evaluasi *Prototyping*
Pada tahapan ini, *prototype* yang sudah dibangun sesuai dengan keinginan pelanggan maka dilakukan evaluasi. Jika sudah sesuai akan dilanjutkan ke tahap pengkodean (*coding*) namun jika belum sesuai maka akan diperbaiki dengan mengulang kembali tahap 1, tahap 2 dan tahap 3.
- d. Tahap Mengkodekan Sistem
Pada tahapan ini, *prototype* yang dibuat sudah diterima dan disepakati maka dilanjutkan dengan pengkodean dengan bahasa pemrograman yang sesuai.
- e. Tahap Pengujian Sistem
Pada tahapan ini, sistem diuji dengan metode *Black Box Testing*. Tujuannya untuk mengetahui apakah sistem sudah berjalan baik atau belum.
- f. Tahap Evaluasi Sistem
Pada tahapan ini, dilakukan evaluasi untuk mengetahui apakah sistem yang dibangun telah sesuai dengan yang diharapkan. Jika sudah memenuhi dapat dilakukan tahap selanjutnya, namun jika tidak maka perlu dilakukan pengulangan pada tahap 4 dan tahap 5.
- g. Tahap Penggunaan Sistem
Pada tahapan ini, sistem yang dibangun siap digunakan oleh pelanggan setelah tahap pengujian berhasil dan diterima oleh pelanggan.

2.2.10 Metode Pengujian Sistem

Metode *black-box* atau *Behavioral Testing* adalah metode pengujian sistem bertujuan untuk melihat program tersebut apakah sesuai dengan fungsi yang diinginkan program tersebut tanpa mengetahui kode program yang dipakai[5]. Keuntungan dari metode ini yaitu penguji tidak harus memiliki pengetahuan tentang program dan penguji juga tidak harus memeriksa keseluruhan kode, dan hanya berfokus pada output yang dihasilkan. Sehingga bertujuan untuk

menunjukkan fungsi dari aplikasi tentang cara bekerja, apakah input dan output sudah sesuai dengan yang diharapkan.

Berikut adalah beberapa teknik yang dapat digunakan, sebagai berikut[5]:

- a. *All Pair Testing*
Dikenal sebagai pairwise testing dan digunakan untuk menguji adanya kemungkinan kombinasi berdasarkan input parameter.
- b. *Boundary Value Analysis*
Berkonsentrasi dalam pencarian *error* dari dalam maupun luar perangkat lunak.
- c. *Cause-effect graph*
Teknik ini menggunakan grafik sebagai dasar dan menggambarkan hubungan antara efek dan penyebab dari *error*.
- d. *Equivalence Partitioning*
Teknik ini membagi data input dari perangkat lunak menjadi beberapa pemisah data.
- e. *Fuzzing*
Teknik ini mencari bug dalam perangkat lunak dengan cara *input* data yang tidak sempurna.
- f. *Orthogonal Array Testing*
Teknik digunakan jika input bernilai kecil, tetapi berat jika digunakan dalam ukuran besar.
- g. *State transition*
Teknik yang dalam bentuk grafik berguna untuk menguji mesin dan navigasi dari UI.