

BAB II

LANDASAN TEORI

2.1. Teori Mengenai Framework dan Bahasa Pemrograman yang Digunakan

2.1.1. HyperText Markup Language (HTML)

HyperText Markup Language (HTML) merupakan sebuah bahasa pemrograman yang digunakan untuk menggambarkan struktur halaman *website* [5]. Susunan bahasa pemrograman HTML terdiri dari perpaduan simbol dan teks yang mengikuti format ASCII (*American Standard Code for Information Interchange*). Dengan adanya kehadiran HTML, sebuah *website* dapat memiliki tampilan yang menarik, interaktif, dan dinamis. Selain itu, bahasa HTML juga dapat dikombinasikan dengan bahasa pemrograman lain seperti Javascript, PHP, dan lain-lain.

2.1.2. Cascading Style Sheet (CSS)

Cascading Style Sheet (CSS) dapat dikatakan sebagai pengatur tampilan elemen dalam sebuah *website* sehingga terlihat lebih terstruktur dan seragam. CSS bukan merupakan bahasa pemrograman layaknya HTML, tetapi keduanya memiliki keterkaitan yang erat [6]. HTML berperan sebagai fondasi utama pada sebuah *website*, sedangkan CSS berperan untuk memperbaiki *style* yang meliputi semua aspek terkait dengan tampilan *website*, mulai dari ukuran gambar, ukuran *border* suatu tabel, jenis dan warna *font* teks, warna tabel, warna *mouse over*, spasi antar paragraf, *alignment*, hingga berbagai parameter lainnya. Keberadaan CSS memungkinkan kita untuk bisa menggunakan berbagai macam pengaturan parameter pada satu halaman *website* sekaligus.

2.1.3. Hypertext Preprocessor (PHP)

Hypertext Preprocessor (PHP) merupakan salah satu bahasa pemrograman yang berjalan dalam sebuah *web server* dan berfungsi sebagai pengolah data pada sebuah *server* [7]. Data yang dikirim oleh *user* nantinya akan diolah dan disimpan pada *database web server* dan

dapat ditampilkan kembali apabila diakses. Bahasa PHP sering digunakan untuk menghasilkan *website* dinamis yang membutuhkan akses basis data (*database*) di dalamnya.

Untuk membuat *website* yang dinamis dan mudah diperbarui setiap saat dari *browser*, dibutuhkan sebuah program yang mampu mengolah data dari komputer *client* atau dari *server* komputer itu sendiri sehingga mudah dan nyaman disajikan di *browser*. Salah satu program yang dapat dijalankan di *server* dan cukup andal adalah PHP sehingga kini penggunaan bahasa PHP dapat dengan mudah ditemukan pada berbagai macam *website* di dunia. Struktur penulisan dari PHP terbagi menjadi dua jenis yaitu *native* dan *framework*. PHP *native* merupakan bentuk penulisan kode untuk perancangan sebuah halaman *website* menggunakan bahasa PHP yang dimulai dari nol, sedangkan *framework* PHP menerapkan prinsip kebalikannya di mana penulisan kode telah memiliki kerangka dan strukturnya tersendiri.

2.1.4. Bootstrap

Bootstrap adalah *front-end framework* atau kerangka kerja berbasis HTML dan CSS yang bersifat terbuka (*open source*) di mana penggunaannya ditujukan untuk kebutuhan tampilan *website* yang memadai [8]. Semenjak tahun 2012, Bootstrap telah semakin diminati karena dilengkapi dengan fitur responsif. Kehadiran fitur responsif dari Bootstrap menjadikan sebuah *website* dapat dilihat dalam berbagai ukuran *device* tanpa mengurangi nilai estetika dari *website* tersebut. Beberapa kemudahan yang disediakan oleh Bootstrap meliputi fleksibilitas pengaturan pada suatu tampilan *website* serta desain yang telah mumpuni dan responsif.

2.1.5. JavaScript

JavaScript merupakan bahasa pemrograman berbasis *prototype* yang digunakan dalam pengembangan sebuah *website*. Secara fungsional, JavaScript digunakan untuk mengubah *website* menjadi lebih interaktif. Contoh sederhana dari implementasi bahasa JavaScript pada *website*

adalah memunculkan halaman *pop-up* ketika membuka sebuah halaman, fungsi validasi *form* sebelum dikirim ke *server*, mengubah bentuk kursor ketika melewati objek tertentu, dan lain-lain [9].

2.1.6. phpMyAdmin

phpMyAdmin adalah sebuah aplikasi atau perangkat bersifat terbuka (*open source*) yang bisa digunakan secara gratis untuk melakukan pemrograman ataupun administrasi pada *database* MySQL. phpMyAdmin sendiri menggunakan bahasa PHP untuk pemrogramannya. Selain itu, phpMyAdmin juga mendukung berbagai operasi MySQL, mulai dari mengelola basis data (*database*), merelasikan tabel-tabel, dan lain-lain. phpMyAdmin tidaklah sama dengan MySQL tetapi keduanya memiliki keterkaitan. phpMyAdmin digunakan sebagai alat yang memudahkan pengoperasian *database* MySQL, sedangkan MySQL adalah *database* itu sendiri, di mana *database* berfungsi sebagai tempat penyimpanan data [10].

2.1.7. Visual Studio Code

Visual Studio Code adalah sebuah program *text editor* hasil rancangan Microsoft di mana fungsi utamanya adalah untuk mendukung pembuatan struktur *code* pada berbagai sistem operasi. Artinya, Visual Studio Code tersedia dan dapat diakses pada berbagai versi sistem operasi mulai dari Linux, Mac, hingga Windows [11]. Visual Studio Code mendukung berbagai bahasa pemrograman seperti JavaScript, TypeScript, Node.js, serta bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang seperti JavaScript, C++, Java, Python, dan lain-lain. Selain itu, Visual Studio Code juga menyediakan banyak fitur menarik yang dapat diintegrasikan ke dalamnya seperti *IntelliSense*, *Git*, *Debugging*, *Integration*, dan fitur ekstensi lainnya yang dapat menambah kemampuan *text editor* ini. Seiring dengan bertambahnya versi aplikasi yang secara berkala diperbarui, fitur-fitur menarik pada Visual Studio Code juga akan terus mengalami penambahan.

2.1.8. Database

Database atau sering juga dikenal sebagai sistem basis data merupakan sebuah sistem yang dihasilkan untuk menyimpan, mengorganisir, hingga menarik data dengan mudah [10]. Adapun beberapa fungsi dari *database* antara lain mempermudah identifikasi data dengan cara pengelompokan data, mempermudah penggunaan *user* dalam berbagai hal, menjadi alternatif terkait masalah penyimpanan ruang dalam suatu aplikasi, dan lain-lain. Dalam kaitannya dengan *website*, *database* berfungsi untuk menyimpan data *website* ke dalam sebuah *server* sehingga dapat diakses pada berbagai perangkat sekaligus.

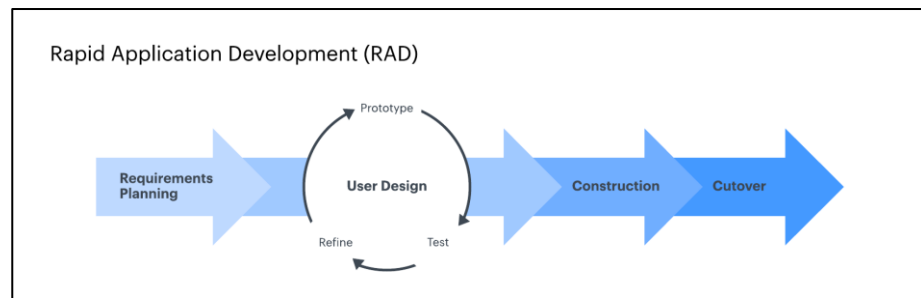
2.1.9. CodeIgniter

CodeIgniter adalah sebuah *framework* pengembangan aplikasi *web* berbasis bahasa pemrograman PHP. Penggunaan CodeIgniter bertujuan untuk menghasilkan *website* dinamis secara efisien. *Framework* ini menggunakan konsep *model, view, controller* yang memungkinkan terjadinya pemisahan komponen pembangun dalam suatu aplikasi [12]. Selain itu, CodeIgniter tergolong memiliki fitur-fitur yang menguntungkan walaupun hanya membutuhkan sedikit konfigurasi [13].

2.2. Teori Mengenai Rancangan Sistem

2.2.1. Rapid Application Development

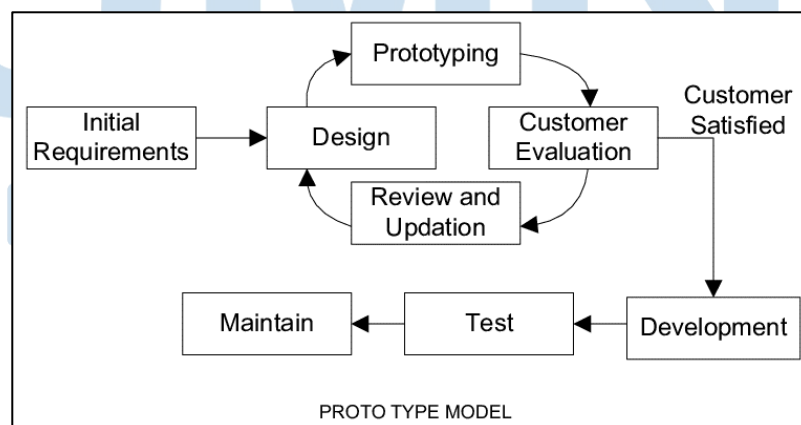
Metode *Rapid Application Development* (RAD) merupakan suatu model pengembangan yang identik dengan durasi pengembangan yang tergolong singkat. Lantaran hal tersebut, metode RAD kerap dijadikan sebagai metode pengembangan sebuah proyek yang mengedepankan aspek kecepatan dan ketepatan sekaligus [14]. Metode RAD terdiri atas empat tahap utama yakni perencanaan kebutuhan sistem (*requirements planning*), desain interaksi antar pengguna (*user design*), pembangunan sistem (*construction*), dan diakhiri dengan proses penyelesaian atau *cutover*. Gambar 2.1 memperlihatkan contoh alur metode pengembangan RAD.



Gambar 2.1 Metode Pengembangan RAD
Sumber: [14]

2.2.2. Prototyping

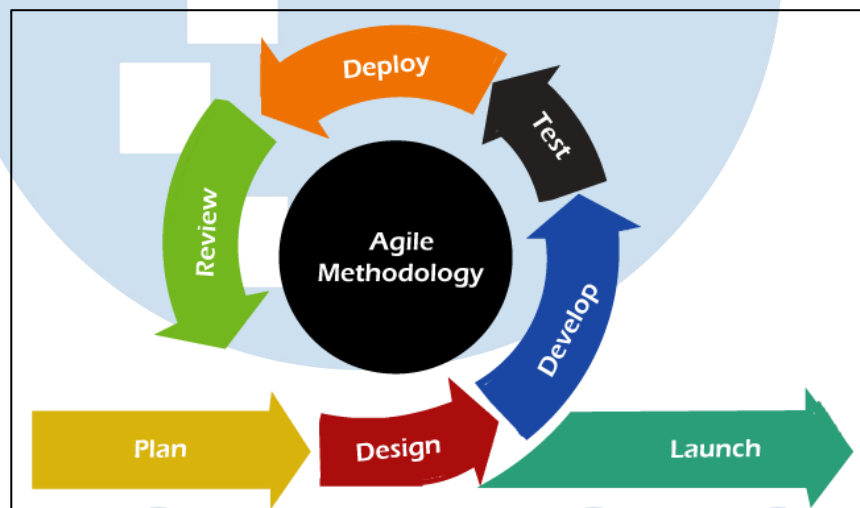
Metode *Prototyping* merupakan metode pengembangan di mana sebuah *prototype* atau gambaran awal mengenai suatu perangkat lunak yang akan dikembangkan dirancang terlebih dahulu [15]. Tujuan yang ingin dicapai dari metode ini yakni peningkatan efisiensi waktu pengerjaan karena proses analisis dan perancangan yang cenderung singkat. Salah satu kelebihan dari metode ini adalah pengetahuan penerima hasil rancangan yang lebih dalam mengenai hasil produk yang diinginkan. Secara keseluruhan, tahap *Prototyping* dimulai dengan pengidentifikasian kebutuhan (*requirements identification*), membangun desain awal *prototype (design)*, pengkodean sistem (*prototyping*), evaluasi desain *prototype (customer evaluation)*, penilaian dan pembaruan *prototype (review and updation)*, pengembangan sistem lebih lanjut (*development*), pengujian seluruh sistem (*testing*), dan diakhiri dengan pemeliharaan sistem (*maintain*). Gambar 2.2 memperlihatkan contoh alur metode pengembangan *Prototyping*.



Gambar 2.2 Metode Pengembangan *Prototyping*
Sumber: [15]

2.2.3. Agile

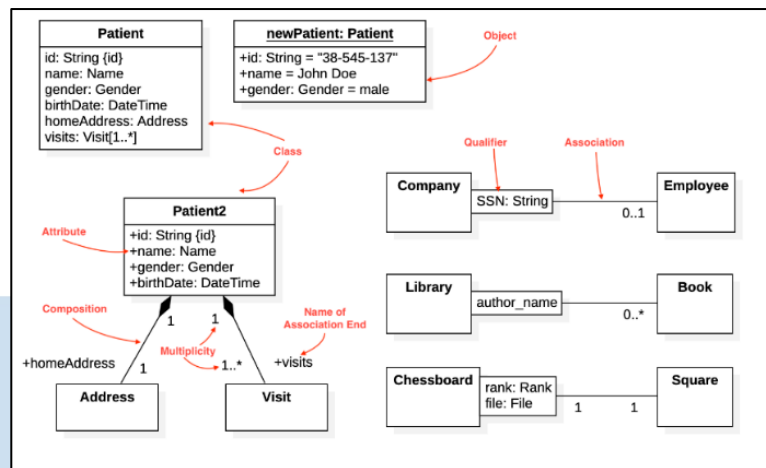
Metode *Agile* dapat dikatakan sebagai model pengembangan yang berbasis fleksibilitas. Pada mekanisme model *Agile*, segala perbaikan atau pembaruan untuk setiap proses pengembangan dapat langsung dilakukan ketika dibutuhkan. Metode *Agile* sendiri terbagi lagi ke dalam beberapa model pendekatan seperti Scrum, Extreme Programming (XP), Spiral, dan lainnya [16]. Masing-masing model pendekatan tersebut dapat disesuaikan dengan kebutuhan organisasi atau perusahaan yang melakukan pengembangan proyek. Gambar 2.3 memperlihatkan contoh alur metode *Agile*.



Gambar 2.3 Metode Pengembangan *Agile*
Sumber: [16]

2.2.4. Class Diagram

Class diagram merupakan sebuah jenis pemodelan pada UML berbentuk diagram statis yang berfungsi untuk menggambarkan struktur pemodelan logis dari suatu sistem. Dengan kata lain, *class diagram* digunakan dalam mendefinisikan bagaimana susunan komponen pada sebuah sistem yang dirancang dapat terhubung dan berinteraksi satu sama lain. Sebuah *class diagram* dilambangkan dengan kotak berisikan atribut tertentu yang kemudian dihubungkan dengan garis yang disebut asosiasi [17]. Gambar 2.4 memperlihatkan salah satu contoh *class diagram*.



Gambar 2.4 Contoh *Class Diagram*

Sumber: [17]

Class diagram memiliki sejumlah komponen yang memiliki fungsi dan arti masing-masing. Komponen pada *class diagram* dapat dilihat melalui tabel 2.1.

Tabel 2.1 Tabel Notasi *Class Diagram*

Simbol	Notasi	Keterangan
	<i>Class</i>	Sebuah <i>class</i> dapat didefinisikan sebagai komponen utama pembangun sistem. Terdapat tiga bagian dari sebuah <i>class</i> . Bagian atas menunjukkan nama dari <i>class</i> , bagian tengah menunjukkan <i>property</i> atau atribut apa saja yang terdapat pada <i>class</i> , dan bagian bawah menunjukkan <i>method-method</i> dari sebuah <i>class</i> .
	<i>Association</i>	<i>Association</i> yang digambarkan dengan sebuah garis lurus digunakan untuk menunjukkan hubungan umum antar <i>class</i> .
	<i>Composition</i>	<i>Composition</i> yang dilambangkan dengan simbol pada kolom notasi bertujuan untuk mendefinisikan hubungan dari sebuah <i>class</i> yang tidak bisa

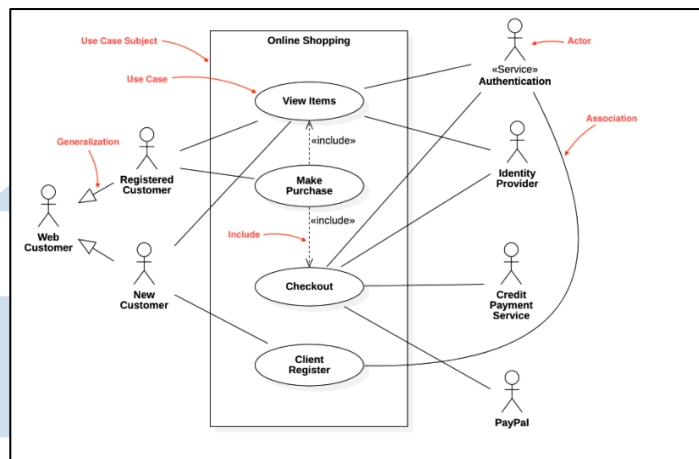
Simbol	Notasi	Keterangan
	<i>Dependency</i>	<i>Dependency</i> merupakan sebuah notasi untuk menunjukkan bahwa sebuah <i>class</i> memerlukan <i>class</i> lain untuk dapat beroperasi semestinya.
	<i>Aggregation</i>	<i>Aggregation</i> yang dilambangkan dengan simbol pada kolom notasi bertujuan untuk menunjukkan bahwa sebuah <i>class</i> merupakan bagian dari <i>class</i> lainnya.
	<i>Generalization</i>	<i>Generalization</i> adalah sebuah notasi untuk menunjukkan relasi <i>inheritance</i> dari <i>child class</i> ke <i>parent class</i> .

Sumber: [18]

2.2.5. Use Case Diagram

Use case diagram sebagai salah satu jenis pemodelan penting lainnya dalam UML merupakan sebuah pemodelan berbentuk diagram yang menjelaskan hubungan antara aktor yang terlibat dalam jalannya suatu sistem dengan sistem itu sendiri. Sederhananya, sebuah *use case diagram* sebagai dasar dari perencanaan sebuah sistem berfungsi untuk mendeskripsikan interaksi apa saja yang terjadi antara para pengguna sistem dengan sistem tersebut [19]. Dengan menggambarkan bentuk interaksi aktor melalui *use case diagram*, proses aktivitas yang ada beserta aktornya dalam sistem dapat digambarkan secara lebih jelas.

Gambar 2.5 memperlihatkan contoh penggunaan *use case diagram* dalam menggambarkan alur sebuah sistem.



Gambar 2.5 Contoh *Use Case Diagram*

Sumber: [19]

Use Case Diagram memiliki sejumlah komponen yang memiliki fungsi dan arti masing-masing. Komponen pada *use case diagram* dapat dilihat melalui tabel 2.2.

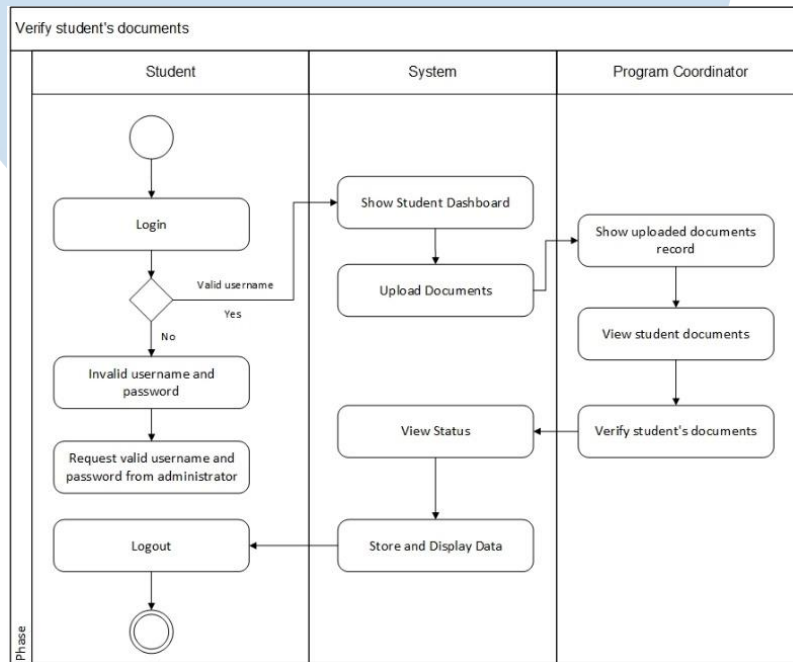
Tabel 2.2 Tabel Notasi *Use Case Diagram*

Simbol	Notasi	Keterangan
	<i>Actor</i>	<i>Actor</i> dapat didefinisikan sebagai pengguna sistem yang berinteraksi dengan <i>use case</i> .
	<i>Use Case</i>	<i>Use case</i> berfungsi untuk menunjukkan deskripsi urutan aksi dalam sistem yang terhubung dengan aktor.
	<i>Association</i>	<i>Association</i> yang memiliki simbol berupa garis lurus digunakan untuk menunjukkan hubungan antara aktor dan <i>use case</i> .
	<i>Dependency</i>	<i>Dependency</i> menunjukkan adanya ketergantungan antara satu elemen dengan elemen lainnya.
	<i>Generalization</i>	<i>Generalization</i> adalah sebuah notasi untuk menunjukkan adanya hubungan antara <i>child object</i> dan <i>parent object</i> .
	<i>System</i>	<i>System</i> berfungsi untuk menandakan bahwa sejumlah objek berada pada area sistem yang sama.

Sumber: [20]

2.2.6. Activity Diagram


Activity diagram atau diagram aktivitas merupakan diagram yang mampu memodelkan rangkaian proses atau alur kerja yang terdapat pada sebuah sistem berdasarkan pihak penggunanya [21]. Terdapat tiga tujuan yang ingin dicapai dari pembuatan *activity diagram*, mulai dari menggambarkan aktivitas yang terdapat pada rancangan sistem, menjelaskan cara kerja sistem secara fungsional, serta mendeskripsikan keterlibatan aktor pada sistem [22]. Umumnya rangkaian aktivitas pada *activity diagram* digambarkan dengan posisi vertikal. Gambar 2.6 memperlihatkan contoh *activity diagram* mengenai alur proses verifikasi dokumen mahasiswa yang melibatkan tiga pihak sekaligus.

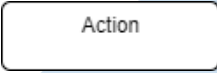




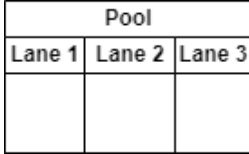


Gambar 2.6 Contoh *Activity Diagram*
Sumber: [22]

Activity Diagram memiliki sejumlah komponen yang memiliki fungsi dan arti masing-masing. Komponen pada *activity diagram* dapat dilihat melalui tabel 2.3.

Tabel 2.3 Tabel Notasi *Activity Diagram*

Simbol	Notasi	Keterangan
	<i>Activity</i>	<i>Activity</i> berisikan deskripsi mengenai interaksi atau aktivitas

Simbol	Notasi	Keterangan
		apa saja yang dilakukan oleh suatu pihak.
	<i>Action</i>	<i>Action</i> merupakan suatu kondisi pada sistem yang menandakan proses eksekusi dari sebuah aksi.
	<i>Initial Node</i>	<i>Initial node</i> menjadi simbol dimulainya suatu objek atau proses.
	<i>Activity Final Node</i>	<i>Activity final node</i> menjadi simbol berakhirnya suatu proses.
	<i>Decision</i>	<i>Decision</i> digunakan untuk menunjukkan kondisi yang memerlukan sebuah pengambilan keputusan atau tindakan guna mencapai kondisi tertentu.
	<i>Line Connector</i>	<i>Line connector</i> yang memiliki simbol berupa anak panah berfungsi untuk menunjukkan koneksi antara setiap objek.
	<i>Swimlane</i>	<i>Swimlane</i> berfungsi untuk memisahkan setiap pihak yang bertanggung jawab atas sebuah aktivitas.

Sumber: [23]

2.2.7. User Acceptance Test

User Acceptance Test (UAT) adalah sebuah proses yang dijalankan dalam upaya memverifikasi bahwa solusi yang dihasilkan dari rancangan sebuah sistem telah berjalan semestinya untuk *user* [24]. Fokus utama pelaksanaan UAT bukanlah untuk menguji apakah suatu sistem dapat berjalan lancar tanpa adanya *system crash* atau kegagalan sistem, melainkan untuk memastikan bahwa pengguna benar-benar menerima solusi dari sebuah tujuan yang ingin dicapai dari rancangan sistem. UAT terdiri dari beberapa jenis seperti *Alpha and Beta Testing*, *Regulation Acceptance Testing*, *Black Box Testing*, dan lainnya [25].

2.2.8. Black Box Testing

Black Box Testing merupakan salah satu jenis UAT yang difungsikan untuk melakukan pengujian terhadap rancangan suatu sistem [26]. Proses menjalankan *Black Box Testing* tidak memerlukan pengetahuan khusus mengenai struktur internal sebuah rancangan sistem sehingga penguji yang terlibat tidak harus berasal dari pihak pengembang atau *developer* sistem [27]. Pengujian rancangan sistem menggunakan metode *Black Box Testing* dapat dikategorikan sebagai tahap finalisasi perancangan sistem yang penting untuk dilakukan guna menghindari kegagalan suatu fitur fungsional [28].

2.3. Penelitian Terdahulu

Tabel 2.4 Penelitian Terdahulu

No.	Nama Peneliti	Jurnal, Volume, Tahun	Judul Artikel	Hasil Penelitian
1	Riswanda, Kusumawati	Jurnal SPIRIT, Vol. 11 (1), Mei 2019	“Rancang Bangun Website Penyedia Lowongan Kerja Disabilitas Kabupaten Pasuruan” [29]	Penelitian yang menggunakan metode pengembangan <i>Waterfall</i> ini menghasilkan sebuah <i>website</i> sebagai platform penyedia lowongan kerja disabilitas. Dengan mengimplementasikan bahasa pemrograman PHP, <i>website</i> yang dihasilkan mampu mencocokkan pekerjaan bagi para penyandang disabilitas sesuai dengan kapabilitas masing-masing.
2	Difla Iklila, Masnur	Jurnal SINTAKS LOGIKA, Vol. 1 (2), Mei 2021	“Sistem Informasi Penyedia Lowongan Kerja Berbasis Web” [30]	Penelitian ini menggunakan metode eksperimen dalam merancang <i>website</i> lowongan kerja serta metode <i>Black Box</i> dan <i>White Box Testing</i> untuk melakukan pengujian terhadap hasil rancangan sistem. Dengan mengimplementasikan

No.	Nama Peneliti	Jurnal, Volume, Tahun	Judul Artikel	Hasil Penelitian
				bahasa pemrograman PHP, <i>website</i> didesain dan dirancang agar bisa menjalankan beberapa fitur-fitur dasar pada sebuah <i>website</i> pencarian kerja seperti <i>login</i> , <i>user profile</i> , <i>job vacancy</i> , dan <i>job application form</i> .
3	Jansen Wiratama, Hari Santoso, Sobiyanto	Computatio: Journal of Computer Science and Information Systems, Vol. 3 (1), 2019	“Sistem Informasi Akademik Berbasis Web Dengan Konsep Human and Computer Interaction: Studi Kasus Stmik Dharma Putra” [31]	Pembuatan prototipe sistem informasi akademik berbasis <i>website</i> pada penelitian ini menggunakan metode pengembangan <i>Waterfall</i> sebagai landasannya. Hasil yang dicapai melalui penelitian ini adalah terciptanya sebuah <i>website</i> yang mampu memudahkan mahasiswa dalam mengisi Kartu Rencana Studi maupun memperlihatkan Kartu Hasil Studi milik mahasiswa. Proses pengujian sistem juga dilakukan dengan cara melakukan <i>Focus Group Discussion</i> , <i>User Acceptance Test</i> , serta perhitungan nilai untuk setiap faktor yang mempengaruhi kualitas dan kecepatan sistem.
4	Baiq Ayu Indah, Amri Muliawan Nur, Imam Fathurrahman	Infotek: Jurnal Informatika dan Teknologi, Vol. 5 (1), Januari 2022	“Aplikasi Penyedia Lowongan Kerja Di Wilayah NTB Berbasis Web Menggunakan Framework Laravel Dan Mysql”[32]	Di dalam penelitian ini, terdapat pengimplementasian <i>framework</i> Laravel serta bahasa pemrograman PHP dan MySQL. Dengan menggunakan metode pengembangan <i>Waterfall</i> , penelitian ini menghasilkan

No.	Nama Peneliti	Jurnal, Volume, Tahun	Judul Artikel	Hasil Penelitian
				sebuah sistem berbasis <i>website</i> yang memiliki sejumlah fitur mulai dari pencarian lowongan kerja berdasarkan kategori tertentu, <i>job applications</i> yang ditampilkan kepada pihak perusahaan penyedia lowongan kerja, fitur daftar lowongan kerja yang dapat diakses oleh para pencari kerja, <i>login</i> , hingga fitur kontak untuk menyediakan layanan pemberian pertanyaan dan saran kepada pihak pengelola <i>website</i> .
5	Mulia Rahmayu, Rosi Kusuma Serli, Reynold Anthonie	Jurnal SIMETRIS, Vol. 10 (2), November 2019	“Rancang Bangun Sistem Pendaftaran Mahasiswa Magang Pada Program Studi Sarjana Sosiologi FISIP Universitas Indonesia Depok” [33]	Penelitian ini merancang sistem pendaftaran mahasiswa magang berbasis <i>website</i> yang sebelumnya diketahui masih dikelola secara manual. Metode penelitian ini adalah <i>descriptive qualitative</i> melalui proses observasi dan wawancara serta menggunakan <i>system development life cycle Waterfall</i> . Perancangan sistem dimulai dengan menyusun alur pemecahan masalah hingga melakukan proses <i>testing</i> menggunakan metode <i>Black Box Testing</i> ketika sistem telah berhasil dirancang.

Berdasarkan kelima penelitian terdahulu yang dijelaskan melalui tabel 2.4, topik penelitian mengenai rancang bangun platform *internship job vacancies* berbasis *website* sebagai media rekrutmen mahasiswa magang dapat diambil

dan dilaksanakan. Perbedaan antara penelitian ini dan penelitian-penelitian sebelumnya berada pada target utama pembuatan sistem. Target utama pembuatan sistem *website* ini adalah untuk mempermudah mahasiswa yang sedang mencari lowongan magang dan perusahaan yang sedang menyediakan lowongan magang, sedangkan pada sejumlah referensi penelitian, *website* dibuat dengan tujuan untuk mempermudah pencarian lowongan pekerjaan penuh waktu atau *full-time job* sehingga target utamanya adalah seluruh masyarakat yang sedang ingin melamar pekerjaan penuh waktu.

Kebaruan selanjutnya terletak pada metode SDLC yang digunakan pada penelitian ini. Sejumlah penelitian yang diuraikan pada tabel 2.4 menggunakan metode SDLC *Waterfall* dalam menghasilkan rancangan *website*, sedangkan penelitian ini menggunakan metode SDLC *Prototyping*. Selain itu, belum terdapat penelitian yang secara khusus menggunakan *framework* CodeIgniter dalam merancang sistem *website* sebagaimana dipergunakan pada penelitian ini.

