

## BAB II

### TINJAUAN PUSTAKA DAN LANDASAN TEORI

#### 2.1 Tinjauan Pustaka

Penelitian terkait sistem informasi berbasis *website* maupun *mobile* sudah banyak dilakukan dan diterapkan di berbagai bidang termasuk dalam bidang pariwisata. Pada penelitian-penelitian sebelumnya yang menggunakan metode *Rapid Application Development* (RAD) akan dijadikan dasar untuk penelitian agar menjadi lebih baik lagi. Penelitian terdahulu yang menurut penulis relevan dengan penelitian yang dilakukan dapat dilihat pada Tabel 2.1.

Tabel 2.1 Tinjauan Pustaka

No.	Peneliti	Judul	Metode	Hasil
1.	Rudi Budi Agung (2019) [9].	Prototipe Sistem Informasi Pemetaan Objek Pariwisata Menggunakan <i>Framework Code Igniter</i> di Propinsi Jakarta.	<i>Rapid Application Development</i> (RAD)	Berdasarkan hasil pengujian terhadap fungsionalitas sistem, dapat dianalisa bahwa aplikasi yang dibuat telah memenuhi kebutuhan para pengguna untuk mencari lokasi pariwisata di Provinsi daerah khusus ibukota Jakarta dengan menampilkan peta objek wisata dalam bentuk <i>google maps</i> dan menampilkan titik

No.	Peneliti	Judul	Metode	Hasil
				koordinat tata letak lokasi objek wisata beserta data atributnya secara lengkap.
2.	Ahmad Apandi (2023) [8].	Pembuatan <i>Website</i> Sistem Informasi Objek Wisata Menggunakan Pendekatan <i>Object Oriented Analysis And Design</i> (OOAD)	<i>Rapid Application Development</i> (RAD)	Berdasarkan hasil penelitian, dapat disimpulkan bahwa sistem informasi objek wisata telah memberikan hasil validasi sesuai dengan kebutuhan pengguna yaitu memberikan informasi mengenai objek wisata di Jakarta Pusat, seperti wisata sejarah, wisata belanja, wisata kuliner dan wisata akomodasi.
3.	Wahyu P. A., Nurudin Santoso & Bayu Priyambadha (2019) [7].	Pengembangan Sistem Informasi Portal Banyuwangi Festival Berbasis <i>Website</i> Menggunakan Metode <i>Rapid Application Development</i> (RAD)	<i>Rapid Application Development</i> (RAD)	Hasil pengujian unit terhadap setiap sampel bernilai 100%, pengujian integrasi pada setiap unit menghasilkan nilai 100% dan pengujian validasi terhadap sistem telah sesuai dengan kebutuhan dengan nilai 100% valid serta pengujian <i>compatibility</i> dapat disimpulkan

No.	Peneliti	Judul	Metode	Hasil
				bahwa sistem dapat berjalan pada beberapa perangkat <i>computer</i> ataupun <i>mobile</i> .
4.	Arliyana dan Noorma Maulidina (2019) [10].	Pusat Informasi Panduan Pariwisata di Kalimantan Tengah Menggunakan Metode <i>Rapid Application Development</i> (RAD)	<i>Rapid Application Development</i> (RAD)	Hasil uji coba pengguna terhadap sistem untuk menilai tampilan dan kemudahan dalam pengoperasian sistem aplikasi menunjukkan sebanyak 62% responden setuju aplikasi yang dibangun bersifat <i>user friendly</i> dan mudah untuk digunakan dalam penyebaran informasi tentang wisata di Kalimantan Tengah.
5.	Fransiskus Lesomar, Hans Wowoe dan Virginia Tulenan (2015) [11].	Rancang Bangun Portal Web Pariwisata Maluku Tenggara	<i>Rapid Application Development</i> (RAD)	Hasil pengujian Portal <i>Web</i> Pariwisata dengan metode RAD menunjukkan hasil yang baik dengan terpenuhinya kebutuhan fungsional sistem setelah dilakukan uji coba oleh pengguna, maka tujuan awal penelitian telah terpenuhi yaitu merancang bangun portal web pariwisata Maluku Tenggara yang dapat

No.	Peneliti	Judul	Metode	Hasil
				menyediakan informasi tempat wisata beserta informasi yang berkaitan.
6.	Tomi Mulyana (2021) [12].	Perancangan Sistem Informasi Paket Wisata Situ Cibereum Kamojang Berbasis <i>Website</i>	<i>Rapid Application Development</i> (RAD)	Hasil dari pengujian menunjukkan bahwa kebutuhan fungsional sistem berhasil terpenuhi setelah dilakukan testing pada <i>website</i> sistem informasi paket wisata Situ Cibereum Kamojang, seperti tampilan <i>Home</i> , profil, <i>login</i> dan buku tamu.
7.	Muchamad Arief Rahman (2015) [2].	Rancang Bangun Sistem Informasi Spasial Berbasis <i>Web</i> Persebaran Potensi Pariwisata Di Kabupaten Bogor	<i>Rapid Application Development</i> (RAD)	Hasil dari pengujian yang dilakukan menggunakan <i>black box testing</i> menunjukkan status “berhasil” pada setiap unit program dari sistem. Dengan begitu kebutuhan fungsional sistem telah terpenuhi, maka tujuan awal penelitian yaitu merancang sistem informasi spasial berbasis <i>web</i> untuk pariwisata di Kabupaten Bogor telah berhasil terpenuhi.

No.	Peneliti	Judul	Metode	Hasil
8.	Endang Sunarya (2017) [13].	Rancang Bangun Sistem Informasi Spasial Persebaran Pantis Sosial Berbasis Web (Studi Kasus: DKI Jakarta)	<i>Rapid Application Development</i> (RAD)	Penelitian ini menghasilkan perancangan sistem informasi spasial berbasis <i>web</i> yang dapat memberikan informasi Pantis Sosial bagi masyarakat secara <i>online</i> . Dari hasil pengujian menggunakan <i>black box testing</i> pada setiap unit program menunjukkan bahwa kebutuhan fungsional sistem berfungsi dengan baik.

Berdasarkan penelitian-penelitian terdahulu, dapat disimpulkan bahwa penelitian ini menggunakan metode *Rapid Application Development* (RAD) yang memiliki perbedaan dari penelitian sebelumnya yaitu dilihat dari lokasi penelitian dan analisa kebutuhan yang menyesuaikan dengan kebutuhan pengguna. Lokasi pada penelitian ini yaitu wisata-wisata yang ada di Cilacap bagian barat. Kebutuhan dari sistem informasi *website* pada penelitian ini yaitu adanya fitur *user* dan admin. *Website* pada penelitian ini akan dimodelkan dengan menggunakan *Unified Modelling Language* (UML) yang terdiri dari *use case* diagram, *activity* diagram dan *sequence* diagram. Kemudian, sistem akan diimplementasikan ke dalam bahasa pemrograman PHP dengan *framework Laravel* dan menyimpan data menggunakan *database MySQL*. Sistem informasi daya tarik wisata pada penelitian ini akan diuji menggunakan pengujian *blackbox testing* untuk menguji fungsionalitas *website* agar berjalan sesuai dengan analisa kebutuhan serta menguji tingkat *usability website* menggunakan pengujian *System Usability Scale* (SUS).

## **2.2 Landasan Teori**

Dasar teori memuat tentang pengetahuan yang berkaitan dengan penelitian yang dilakukan untuk membantu peneliti dalam melakukan penelitian. Berikut ini merupakan dasar teori yang berkaitan dengan penelitian ini.

### **2.2.1 Rancang Bangun**

Perancangan merupakan serangkaian langkah yang menerjemahkan hasil analisis sistem ke dalam bahasa pemrograman dan menjelaskan secara rinci bagaimana komponen sistem akan diimplementasikan. Sedangkan pengertian pengembangan sistem adalah kegiatan menciptakan sistem baru atau mengganti atau menyempurnakan seluruh atau sebagian sistem yang sudah ada [13].

### **2.2.2 Website**

*Website* merupakan suatu aplikasi yang menggunakan protokol HTTP (*Hypertext Transfer Protocol*) dan berisi dokumen multimedia (teks, gambar, animasi, video) yang dapat diakses melalui perangkat lunak yang disebut *browser*. *Website* adalah cara yang cukup efektif untuk memperoleh informasi atau melakukan transaksi barang atau jasa [3].

### **2.2.3 Sistem Informasi**

Sistem Informasi adalah sekumpulan fungsi yang saling berinteraksi dan membentuk satu kesatuan untuk melakukan kegiatan pengolahan informasi, termasuk menerima masukan berupa data, mengolah dan menghasilkan keluaran berupa informasi yang berguna bagi penerima atau pengguna dan dapat membantu dalam pengambilan keputusan untuk pengguna [14].

### **2.2.4 Rapid Application Development (RAD)**

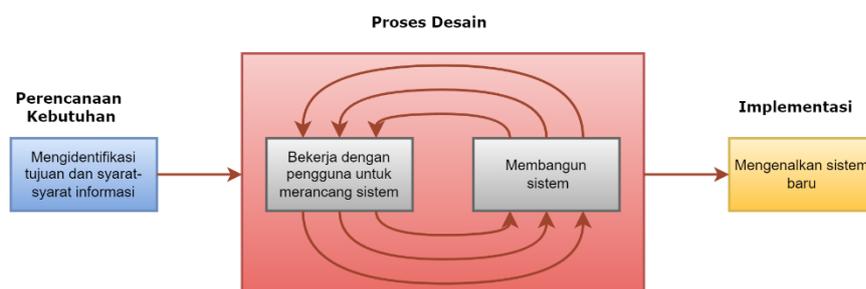
*Rapid Application Development (RAD)* merupakan salah satu metode dengan pendekatan berorientasi objek terhadap pengembangan sistem. RAD menerapkan strategi manajemen proyek secara cepat dalam proses pembangunan perangkat lunak [15]. Hal tersebut dikarenakan RAD berfokus pada pembangunan prototipe. Dengan menekankan iterasi pada perancangan prototipe, metode RAD memungkinkan penyelesaian pembangunan secara cepat dari suatu sistem yang dapat berkembang menjadi suatu sistem akhir karena manajer proyek dan

pemangku kepentingan dapat mengukur kemajuan proyek serta berkomunikasi secara langsung untuk membahas permasalahan atau perubahan sistem [7]

Kelebihan dalam menggunakan metode RAD antara lain:

1. Mudah untuk diamati karena menggunakan model prototipe, sehingga *user* lebih mengerti akan sistem yang dikembangkan.
2. Perubahan sistem yang dirasa perlu oleh *user* ketika sedang berjalan mudah untuk diakomodasi.
3. Memiliki kemampuan untuk menggunakan kembali komponen yang sudah ada sehingga pengembang tidak perlu memulai dari awal kembali dan waktu menjadi lebih singkat [4].

RAD menggunakan pendekatan berulang (*iterative*) pada saat pengembangan sistem. Dalam metode ini, model kerja sistem dibuat pada awal tahap pengembangan untuk mengidentifikasi kebutuhan pengguna, dan kemudian dihapus. Sistem informasi pada umumnya membutuhkan waktu setidaknya 180 hari untuk dikembangkan, namun dengan menggunakan teknik RAD, Anda dapat menyelesaikan sistem dalam waktu 30 hingga 90 hari. Proses dalam menggunakan metode *Rapid Application Development* (RAD) dibagi dalam 3 fase yaitu, tahap perencanaan kebutuhan, proses desain dan implementasi [15]. Fase – fase tersebut dapat dilihat pada Gambar 2.1.



Gambar 2.1 Tahapan *Rapid Application Development* (RAD) [15]

Agar lebih mudah dipahami, berikut adalah penjelasan mengenai fase pada pengembangan perangkat lunak dengan metode RAD.

## **1. Fase Perencanaan Kebutuhan**

Tahap ini merupakan tahap awal dalam suatu pengembangan sistem dengan melakukan identifikasi masalah dan pengumpulan data untuk menentukan tujuan dan batasan sistem serta alternatif pemecahan masalah. Analisis dari perencanaan kebutuhan digunakan untuk mengetahui aktivitas apa saja yang akan ada dalam sistem [15].

## **2. Fase Proses Desain**

Pada tahap ini proses desain mulai dilakukan. Keterlibatan pengguna pada tahap sebelumnya sangatlah penting agar desain yang dibangun sesuai dengan kebutuhan pengguna. Pada tahap ini pengguna juga merespon *prototype* yang telah dirancang, apabila masih ada ketidaksesuaian desain terhadap kebutuhan pengguna maka akan dilakukan perbaikan desain secara berulang. *Output* dari tahapan ini adalah spesifikasi *software* yang meliputi organisasi di dalam sistem, struktur data dan lain-lain [15].

## **3. Fase Implementasi**

Tahap ini merupakan tahapan desain dari suatu sistem yang telah disetujui pada tahapan sebelumnya akan dibangun. Segera setelah sistem dibangun maka akan dilakukan proses pengujian untuk mengetahui apakah terdapat kesalahan pada sistem yang dikembangkan [15]. Pada penelitian ini proses pengujian dilakukan menggunakan metode *blackbox testing* dan metode *System Usability Scale (SUS)*.

### **2.2.5 Metode Pengujian Sistem**

#### **1. Pengujian *Blackbox Testing***

*Blackbox testing* merupakan salah satu metode yang mudah digunakan. Pengujian ini berfokus pada fungsionalitas perangkat lunak. Tujuan dari penggunaan metode *blackbox testing* adalah untuk mengetahui kelemahan dari sistem data yang dihasilkan, apakah data yang dimasukkan akan sesuai dengan data setelah dieksekusi serta menghindari kesalahan sebelum digunakan *user* [16], [17].

#### **2. Pengujian *System Usability Scale (SUS)***

Metode SUS merupakan salah satu metode yang dapat digunakan untuk mengukur tingkat *usability* pada suatu *website*. Metode ini pertama kali dikemukakan oleh J Brooke pada tahun 1986. Metode SUS menggunakan kuisioner

yang terdiri dari 10 pertanyaan dengan jawaban berupa skala likert dengan 5 skor jawaban mulai dari skala “Sangat Tidak Setuju” hingga skala “Sangat Setuju” pada setiap pertanyaannya [18], [19]. Kesepuluh pertanyaan dalam kuisisioner SUS serta 5 skor jawabannya dapat dilihat pada Tabel 2.2 dan 2.3.

Tabel 2.2. Daftar Pertanyaan Kuisisioner SUS [18]

No.	Pertanyaan SUS
1.	Saya berpikir akan menggunakan <i>website</i> sistem informasi daya tarik wisata lagi.
2.	Saya merasa <i>website</i> sistem informasi daya tarik wisata ini rumit untuk digunakan
3.	Saya merasa <i>website</i> sistem informasi daya tarik wisata ini mudah untuk digunakan.
4.	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan <i>website</i> daya tarik wisata ini.
5.	Saya merasa fitur-fitur dalam <i>website</i> daya tarik wisata ini berjalan dengan semestinya.
6.	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada <i>website</i> daya tarik wisata ini.
7.	Saya merasa orang lain akan memahami cara menggunakan <i>website</i> ini dengan cepat.
8.	Saya merasa <i>website</i> ini membingungkan
9.	Saya merasa tidak ada hambatan dalam menggunakan <i>website</i> ini.
10.	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan <i>website</i> ini.

Tabel 2.3. Daftar Skor Jawaban Kuisisioner SUS [18]

Skor	Jawaban
1	Sangat Tidak Setuju (STS)
2	Tidak Setuju (TS)

Skor	Jawaban
3	Netral (N)
4	Setuju (S)
5	Sangat Setuju (SS)

### Aturan Menghitung SUS

*System Usability Sacle* (SUS) memiliki beberapa aturan dalam perhitungan skor SUS. Aturan-aturan saat perhitungan skor pada kuesioner SUS [19]:

1. Setiap pertanyaan bernomor ganjil, skor setiap pertanyaan yang didapat dari skor pengguna akan dikurangi 1.
2. Setiap pertanyaan bernomor genap, skor akhir didapat dari nilai 5 dikurangi skor pertanyaan yang didapat dari pengguna.
3. Skor SUS didapat dari hasil penjumlahan skor setiap pertanyaan yang kemudian dikali 2,5.

Aturan perhitungan skor berlaku pada 1 responden. Selanjutnya, skor SUS dari setiap responden dicari skor rata-rata dengan menjumlahkan semua skor dan dibagi dengan jumlah responden. Rumus menghitung skor SUS [19]:

$$\bar{x} = \frac{\sum x}{n}$$

$\bar{x}$  = Skor rata – rata

$\sum x$  = Jumlah skor SUS

n = Jumlah responden [19].

Sebuah sistem dapat dikatakan memiliki tingkat *usability* yang baik apabila memiliki nilai *System Usability Scale* (SUS) di atas 70 [20].

Tabel 2.4. Rentang Nilai SUS [20]

SUS Score	Grade	Adjective Rating
> 80.3	A	<i>Excellent</i>
68 – 80.3	B	<i>Good</i>

<b>SUS Score</b>	<b>Grade</b>	<b>Adjective Rating</b>
68	C	<i>Ok</i>
51 – 68	D	<i>Poor</i>
< 51	F	<i>Awful</i>

### 2.2.6 Unified Modelling Language (UML)

*Unified Modelling Language* merupakan salah satu pemodelan visual yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak yang berorientasikan pada objek. UML memiliki standar penulisan atau semacam *blueprint* dimana memuat sebuah proses bisnis, penulisan kelas – kelas dalam sebuah bahasa yang spesifik [21].

Terdapat beberapa diagram pada UML yang digunakan untuk merancang dan mengimplementasikan sebuah sistem perangkat lunak, diantaranya [22]:

#### 1. Use Case Diagram

*Use Case* diagram adalah gambaran dari fungsionalitas yang diharapkan saat merepresentasikan sebuah sistem. *Use Case* diagram digunakan untuk mengetahui fungsi apa saja yang ada dalam sistem perangkat lunak dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Komponen dari *Use Case* diagram terdiri dari: *actor*, *Use Case* dan *relation*. *Actor* berperan sebagai pemain, sedangkan *Use Case* adalah apa yang dilakukan *actor* dengan *relation* sebagai penunjuknya [23].

Tabel 2.5 Simbol *Use Case* Diagram [23].

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

## 2. Activity Diagram

*Activity* diagram adalah diagram yang menggambarkan tentang aktivitas – aktivitas dalam sebuah sistem. *Activity* diagram menyajikan sebuah mekanisme agar dapat menggambarkan kegiatan yang terlihat secara *parallel* [24]. Tujuan dari *Activity* diagram yaitu menjelaskan urutan aktivitas dalam suatu proses, mudah dalam memahami proses yang ada dalam sistem secara keseluruhan dan mengetahui aktivitas aktor/pengguna berdasarkan *use case* diagram.

Tabel 2.6 Simbol *Activity* Diagram [24].

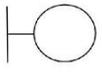
Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan / Decision	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan / Join	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
	Swimlane	Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

## 3. Sequence Diagram

*Sequence* diagram lebih detail dari *activity* diagram karena pada bagian ini menggambarkan bagaimana sebuah objek berinteraksi satu sama lain sepanjang waktu sehingga lebih terperinci [23]. *Sequence* diagram berguna untuk memahami dan merancang sistem yang kompleks karena diagram ini memberikan gambaran yang jekas dan ringkas tentang perilaku dan interaksi sistem. Tujuan dari *sequence* diagram adalah sebagai berikut:

1. Memberikan representasi visual yang jelas dari aliran proses atau sistem.
2. Memodelkan logika dari prosedur, fungsi atau operasi yang kompleks.
3. Melihat bagaimana objek dan komponen yang berbeda berinteraksi satu sama lain untuk menyelesaikan suatu proses.

Tabel 2.7 Simbol *Sequence Diagram* [23].

Gambar	Nama	Keterangan
	Entity Class	Gambaran sistem sebagai landasan dalam menyusun basis data
	Boundary Class	Menangani komunikasi antar lingkungan sistem
	Control Class	Bertanggung jawab terhadap kelas-kelas terhadap objek yang berisi logika
	Recursive	Pesan untuk dirinya
	Activation	Mewakili proses durasi aktivasi sebuah operasi
	Life Line	Komponen yang digambarkan garis putus terhubung dengan objek

### 2.2.7 HTML (*Hyper Text Markup Language*)

HTML merupakan Bahasa pemrograman standar yang digunakan untuk membuat halaman *website* yang dapat diakses untuk menampilkan berbagai informasi di dalam sebuah *browser*. HTML digunakan juga sebagai link antara dokumen dalam situs atau bias dalam *computer* menggunakan *localhost*. HTML berfungsi sebagai pengelola dari serangkaian data dan informasi sehingga suatu dokumen atau *file* dapat diakses dan ditampilkan di internet melalui layanan *website* [25].

### 2.2.8 CSS

CSS merupakan salah satu bahasa desain *website* yang digunakan untuk mengontrol format tampilan sebuah halaman *website* [13]. Biasanya CSS digunakan untuk mendesain halaman HTML dan XHTML, tetapi 16 sekarang CSS dapat digunakan untuk segala dokumen XML, termasuk SVG, XUL dan ANDROID [9]. Fungsi utama CSS adalah untuk memudahkan pemrograman ulang sebuah *website* dengan memisahkan konten utama dengan tampilan dokumen yang meliputi *layout*, warna dan juga *font*. Pemisahan tersebut membuat daya akses

konten pada *web* menjadi lebih fleksibel serta mengurangi kerumitan dalam penulisan kode dan struktur konten [25], [26].

### 2.2.9 PHP (*Hypertext Preprocessor*)

Bahasa pemrograman PHP adalah Bahasa pemrograman untuk membuat halaman *web* yang bersifat *server-side scripting*. *Server-side scripting* adalah pemrosesan data yang dilakukan di sisi *server*. Sederhananya, *server* itu sendiri menerjemahkan *script* program, baru kemudian hasilnya dikirim ke *client* yang membuat permintaan. Pada prinsipnya *server* akan bekerja apabila ada permintaan dari *client*. Dalam hal ini, *client* menggunakan kode PHP untuk mengirim permintaan ke *server*. Sistem kerja dari PHP dimulai dengan permintaan yang berasal dari halaman *website* oleh *browser* [27].

PHP bersifat dinamis dan dapat berjalan di banyak sistem operasi yang berbeda seperti *Windows*, *Linux*, dan *Mac Os*. Selain *Apache*, PHP juga mendukung beberapa *web server* lainnya, seperti *Microsoft ISS*, *Caudium* dan *PWS*. PHP dapat memanfaatkan *database* yang biasa digunakan dengan PHP yaitu *MySQL*. Namun, PHP juga mendukung sistem manajemen *database Oracle*, *Microsoft Access*, *Interbase*, *d-Base* dan *PostgreSQL* [2].

### 2.2.10 *Laravel*

*Laravel* adalah *framework* pengembangan *web MVC* yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan dan pemeliharaan, serta meningkatkan produktivitas kerja dengan sintaks dan fungsionalitas yang jelas serta dapat sangat mengurangi waktu penerapan. *Laravel* adalah *framework* PHP terbaru karena menyediakan PHP 5.3 atau lebih baru yang menekankan pada kesederhanaan dan fleksibilitas dalam desainnya [27].

### 2.2.11 *MySQL*

*MySQL* merupakan *software* yang termasuk sebagai *database server* yang sangat terkenal. Kepopulerannya dikarenakan *MySQL* menggunakan *SQL* sebagai dasar untuk mengakses *database*-nya. *MySQL* juga bersifat multiplatform yaitu dapat dijalankan pada berbagai sistem operasi. *MySQL* juga termasuk jenis *RDBMS (Relational Database Management System)*. Oleh sebab itu, istilah seperti *table*, baris dan kolom digunakan juga pada *MySQL*[2], [28].