

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian dilakukan berdasarkan pada penelitian yang pernah dilakukan sebelumnya. Berikut ini adalah penelitian yang serupa dengan penelitian yang akan dilakukan:

Putu Sidiarta, Anak Agung Ayu Putri Ardyanti, I Gede Juliana Eka Putra (2018), Mahasiswa STMIK Primakara Denpasar Bali, melakukan penelitian tentang Rancang Bangun Sistem Informasi *Marketplace* Penyewaan Lapangan Futsal *Berbasis website*. Sistem diharapkan dapat memudahkan para penyewa lapangan untuk mendapatkan informasi secara *real time* tanpa harus datang langsung atau menanyakan kepada pengelola lapangan yang bersangkutan. Untuk model pengembangan aplikasi menggunakan Model *Waterfall*. Pada sistem ini penyewa tidak dapat mengubah jadwal penyewaan, melakukan pembayaran melalui system, dan tidak memiliki fitur notifikasi. Proses pengujian dilakukan dengan metode *black box*.

Kharis Anwar, Lilik Dwi Kurniawan, M. Ijur Rahman, Nur Ani (2020), Mahasiswa Universitas Mercu Buana Jakarta Barat, melakukan penelitian tentang Aplikasi *Marketplace* Penyewaan Lapangan Olahraga Dari Berbagai Cabang Dengan Metode *Agile Development*. Dari hasil penelitian yang telah dilakukan, dapat ditarik kesimpulan bahwa dengan menggunakan konsep *marketplace* pada sistem informasi penyewaan lapangan olahraga dapat mempertemukan calon penyewa lapangan dengan pengelola lapangan dalam bertukar informasi secara *real-time*. Pada sistem ini penyewa tidak dapat mengubah jadwal penyewaan, pembayaran masih terkendala dalam proses perizinan dan persetujuan dengan pihak pengelola *payment gateway*. Proses pengujian dilakukan dengan metode *black box*.

M. Sabtrio Vie Moro (2019), Mahasiswa Institut Informatika dan Bisnis Darmajaya Bandar Lampung, melakukan penelitian tentang Sistem Informasi *Marketplace* Sarana Olahraga pada Lampung Walk. Dengan adanya sistem informasi untuk pemesanan dan penjadwalan lapangan ini akan memudahkan user dalam melakukan pemesanan lapangan olahraga pada Lampung Walk. Pada

sistem ini penyewa tidak dapat mengubah jadwal penyewaan, pembayaran langsung di transfer ke pemilik Lampung Walk, tidak memiliki fitur notifikasi, dan Sistem hanya digunakan untuk sarana olahraga pada Lampung Walk. Proses pengujian dilakukan dengan metode *black box*.

Tabel 2. 1 Penelitian Terkait

No	Penulis	Judul	Keterangan
1	Putu Sidiarta, Anak Agung Ayu Putri Ardyanti, I Gede Juliana Eka Putra (2018)	Rancang Bangun Sistem Informasi <i>Marketplace</i> Penyewaan Lapangan Futsal <i>Berbasis web</i>	<ul style="list-style-type: none"> - Pembuatan aplikasi ini dilakukan dengan Metode <i>Waterfall</i> - Aplikasi ini diimplementasikan dengan <i>Framework Codeigniter</i>, dan dibuat dengan Bahasa pemograman PHP, serta MySQL sebagai databasanya, dan untuk metode pengujian sistem menggunakan <i>black box testing</i>. - Aplikasi ini dapat digunakan oleh 3 pengguna, yaitu admin, pengelola, dan penyewa
2	Kharis Anwar, Lilik Dwi Kurniawan, M. Ijur Rahman, Nur Ani (2020)	Aplikasi Marketplace Penyewaan Lapangan Olahraga Dari Berbagai Cabang Dengan Metode Agile Development	<ul style="list-style-type: none"> - Dalam proses pembangunan aplikasi dilakukan dengan menggunakan metode pengembangan aplikasi <i>agile method</i>. - Implementasi aplikasi menggunakan <i>framework</i> Laravel, bahasa pemograman PHP, serta <i>Database Management System MySQL</i> - Pengguna yang ada dalam

			aplikasi ini yaitu penyewa, pengelola, dan admin.
3	M. Sabtrio Vie Moro (2019)	Sistem Informasi Marketplace Sarana Olahraga pada Lampung Walk Berbasis Web	<ul style="list-style-type: none"> - Pengguna dari sistem ini adalah admin dan pelanggan - Alat pengembangan sistem yang digunakan adalah SDLC (<i>Waterfall</i>) - Percancangan sistem telah berhasil mengolah data informasi yang ada melalui <i>login</i> administrator

Tabel 2. 2 Penelitian Yang Dilakukan

No	Penulis	Judul	Keterangan
1	Ammar Alvi Hidayatullah	Sistem Informasi Marketplace Penyewaan Lapangan Futsal di Kota Singkawang Berbasis Website	<ul style="list-style-type: none"> - Pengguna sistem yaitu Admin, Pengelola Lapangan, dan Penyewa Lapangan - Sistem yang dibangun berbasis <i>website</i> - Fitur pembayaran menggunakan qr code untuk mempermudah proses pembayaran - Adanya pemberitahuan jika <i>booking</i> dan pembayaran telah berhasil dilakukan - Metode perancangan sistem menggunakan <i>waterfall</i>

2.2 Kajian Pustaka

2.2.1 Futsal

Futsal adalah permainan bola yang dimainkan oleh dua tim, yang masing-masing beranggotakan lima orang. Tujuannya adalah memasukkan bola ke

gawang lawan, dengan memanipulasi bola dengan kaki. Selain lima pemain utama, setiap regu juga diizinkan memiliki pemain cadangan. Tidak seperti permainan sepakbola dalam ruangan lainnya, lapangan futsal dibatasi garis, bukan net atau papan (Maryati, 2012).

2.2.2 *E-Marketplace*

E-Marketplace merupakan media online berbasis internet (*web-based*) tempat melakukan kegiatan bisnis dan transaksi antara pembeli dan penjual. Pembeli dapat mencari *supplier* sebanyak mungkin dengan kriteria yang diinginkan, sehingga memperoleh sesuai harga pasar. Sedangkan bagi *supplier*/penjual dapat mengetahui perusahaan-perusahaan yang membutuhkan produk/jasa mereka.

E-marketplace merupakan lokasi online di mana pembeli dan penjual melakukan transaksi komersial seperti menjual barang, jasa atau informasi (Turban, et al., 2015).

Komponen dari sebuah *marketplace* hampir sama dengan komponen pada pasar tradisional pada umumnya, yang paling penting agar terjadinya sebuah transaksi adalah dengan adanya calon penjual dan pembeli. Disini akan diuraikan beberapa komponen yang menunjang sebuah *marketplace* itu sendiri, yaitu:

1. Pelanggan

Pelanggan berasal dari seluruh dunia, yang *surf* melalui web.

2. Penjual

Jutaan toko ada di web, iklan dan menawarkan barang yang sangat bervariasi

3. Barang dan Jasa

Barang dan jasa mempunyai tipe fisik dan digital. Digital produk ini adalah barang yang diubah menjadi format digital dan dikirim melalui internet.

4. Infrastruktur

Network, hardware, software dan lainnya adalah infrastruktur yang harus disiapkan dalam menjalankan *e-marketplace*.

5. *Front-end*

Penjual dan pembeli berhubungan dalam *marketspace* melalui sebuah *front-end*. *Front-end* ini berisi portal penjual, catalog elektronik, *shopping cart*,

mesin pencari, mesin lelang.

6. *Back-end*

Aktivitas yang berhubungan dengan pemesanan dan pemenuhan pemesanan, manajemen persediaan, pembelian dari pemasok, akuntansi dan finansial, proses pembayaran, pengepakan, dan pengiriman dilakukan di *back-end*.

7. *Intermediaries*

Pihak ketiga yang mengoperasikan antara penjual dan pembeli. Kebanyakan dioperasikan secara komputerasisasi

8. Partner Bisnis Lainnya

Misalnya pengiriman, menggunakan internet untuk berkolaborasi, kebanyakan dengan rantai pemasok.

9. Jasa Pendukung

Jasa sertifikasi, jasa keamanan biasanya masuk dalam jasa pendukung.

2.2.3 Penyewaan

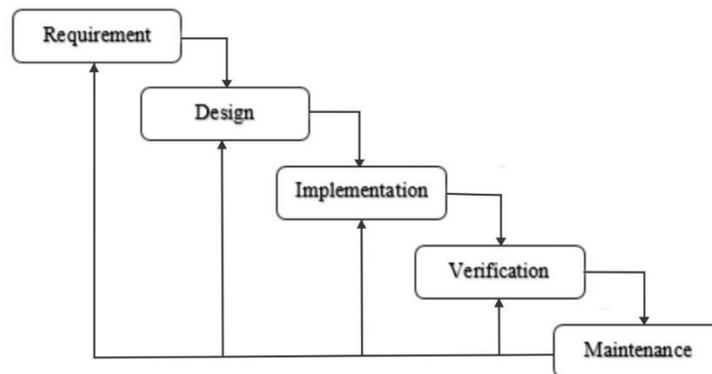
Penyewaan adalah suatu kesepakatan atau persetujuan dimana pihak yang satu menyanggupkan dirinya untuk menyerahkan suatu kebendaan kepada pihak lain agar pihak ini dapat menikmatinya dalam jangka waktu tertentu, yang mana pihak yang belakang ini sanggup membayarnya (Subekti & Tjitrosudibio, 1996).

Penyewaan adalah suatu kegiatan yang melayani jasa peminjaman dengan tidak mengabaikan suatu ketentuan atau kesepakatan dan syarat-syarat yang berlaku di dalam organisasi tersebut guna mencapai satu tujuan bersama. Sewa-menyeewa diatur dalam pasal 1548 KUH Perdata Sewa menyewa adalah suatu persetujuan, yang mana pihak yang satu mengikatkan diri untuk memberikan kenikmatan suatu barang kepada pihak yang lain selama waktu tertentu, dengan pembayaran suatu harga yang disanggupi oleh pihak terakhir itu (Subekti & Tjitrosudibio, 1996).

2.2.4 Metode *Waterfall*

Metode air terjun atau disebut metode *waterfall* sering dinamakan siklus hidup klasik (*classic life cycle*), dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan

Requirement, Design System, Implementation, Verification, Maintenance (Pressman, Rekayasa Perangkat Lunak : Pendekatan Praktisi Jilid 1, 2002).



Gambar 2.1 Tahapan Metode *Waterfall*

Berikut adalah tahapan metode *waterfall*:

1. *Requirement Analysis*

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi ini biasanya dapat diperoleh melalui wawancara, diskusi atau survey langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

2. *Design System*

Spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari dalam fase ini dan dibuat menjadi rancangan sistem. Sehingga terlihat gambaran bentuk sistem yang akan dibuat. Desain dapat berupa bagan (*chart*) yang menunjukkan prosedur sistem secara logika, desain *proses bisnis*, dan desain basis data.

3. *Implementation*

Pada tahap ini merupakan implementasi dari tahap design. Gambaran sistem di tahap sebelumnya di implementasikan dalam bentuk *coding* atau kode-kode program sampai menghasilkan sistem akhir atau sistem jadi.

4. *Verification*

Tahapan ini merupakan tahap dimana sistem atau aplikasi yang telah dibuat di uji. Apakah sistem tersebut layak atau masih harus diperbaiki. Pengujian biasanya dilakukan untuk melihat apakah sistem masih error atau bahkan tidak jalan. Apabila sistem dianggap sudah baik maka sistem itu bisa diterapkan.

5. *Maintenance*

Tahap akhir dalam metode waterfall. Perangkat lunak yang sudah jadi, dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaikan implementasi unit sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

2.2.5 **Alat Bantu Perancangan Sistem**

2.2.5.1 *Data Flow Diagram (DFD)*

DFD merupakan suatu cara atau metode untuk membuat rancangan sebuah sistem yang mana berorientasi pada alur data yang bergerak pada sebuah sistem nantinya. *Data Flow Diagram (DFD)* adalah sebuah alat yang menggambarkan aliran data sampai sebuah sistem selesai, dan kerja atau proses dilakukan dalam sistem tersebut (Indrajani, 2011).

Ada beberapa simbol yang digunakan pada DFD : (Jogiyanto, 2001).

1. Kesatuan Luar (*External Entity*)

Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lain yang berada pada lingkungan luarnya yang memberikan input atau menerima output dari sistem.

2. Arus Data (*Data Flow*)

Arus Data (*data flow*) di DFD diberi simbol suatu panah. Arus data ini mengalir di antara proses, simpan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

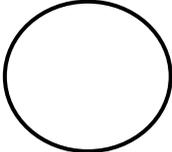
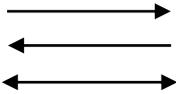
3. Proses (*Process*)

Proses (*process*) menunjukkan pada bagian yang mengubah input menjadi output, yaitu menunjukkan bagaimana satu atau lebih input diubah menjadi beberapa output. Setiap proses mempunyai nama, nama dari proses ini menunjukkan apa yang dikerjakan proses.

4. Simpanan Data (*Data Store*)

Data Store merupakan simpanan dari data yang dapat berupa suatu file atau database pada sistem komputer.

Tabel 2. 3 Simbol DFD

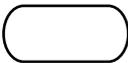
Simbol DFD	Nama
	Entitas / Kesatuan luar
	Proses
	Arus data / Aliran data
	Data store / Simpanan data

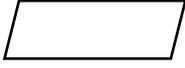
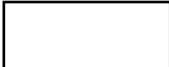
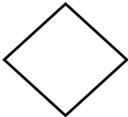
2.2.5.2 Flowchart

Flowchart adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi.

Flowchart merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempermudah penyelesaian masalah yang khususnya perlu dipelajari dan dievaluasi lebih lanjut (Indrajani, 2011).

Tabel 2. 4 Simbol *Flowchart*

Simbol Flowchart	Keterangan
Terminal	Memulai dan Mengakhiri suatu proses
	
Input / Output	Digunakan untuk proses data maupun input

	data
Proses	Menampilkan suatu proses
	
Keputusan	Untuk menyeleksi kondisi didalam program. (percabangan)
	
Persiapan	Digunakan untuk pemberian nilai awal suatu besaran
	
Proses pendefinisian	Menunjukkan suatu operasi yang rinciannya ditunjukkan ditempat lain
	
Garis alir	Garis suatu proses atau menunjukkan arus proses yang berjalan

2.2.5.3 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. ERD menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data dengan menggunakan kardinalitas relasi. Kardinalitas relasi menunjukkan jumlah maksimum tupel yang dapat berelasi dengan entitas pada entitas yang lain (Ladjamudin, 2005).

Terdapat 3 macam kardinalitas relasi yaitu sebagai berikut:

1. *One to One* (1:1)

Setiap anggota entitas A hanya boleh berhubungan dengan satu anggota entitas B, begitu pula sebaliknya.

2. *One to Many* (1:M / *Many*)

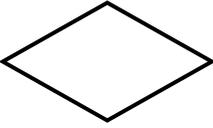
Setiap anggota entitas A dapat berhubungan dengan lebih dari satu

anggota entitas B tetapi tidak sebaliknya.

3. *Many to Many* (M:M)

Setiap entitas A dapat berhubungan dengan banyak entitas himpunan entitas B dan demikian pula sebaliknya.

Tabel 2. 5 Simbol ERD

Simbol ERD	Keterangan
	Entitas : Memberikan identitas yang mewakili suatu objek dan dapat dibedakan dengan objek yang lain.
	Relasi : Menunjukkan hubungan diantara sejumlah entitas yang berbeda.
	Atribut : Properti yang dimiliki oleh suatu entitas, dimana dapat mendeskripsikan karakteristik dari entitas tersebut.
	Alur : Garis yang memiliki fungsi untuk menghubungkan atribut dengan entitas dan entitas dengan relasi.

2.2.5.4 Kamus Data (*Data Dictionary*)

Kamus data merupakan tempat penyimpanan dari aliran-aliran data, *file* dan proses-proses dalam sebuah sistem. Kamus data digunakan untuk menjelaskan semua data yang mengalir atau digunakan dalam sistem, yaitu mengenai arus data yang masuk ke dalam sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.

Kamus Data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. *Data dictionary* tidak menggunakan notasi grafik sebagaimana halnya *data flow diagram*. Kamus data berfungsi untuk membantu user agar dapat mengerti aplikasi secara rinci (Sutabri, 2004).

2.2.6 Tools Pemograman

2.2.6.1 *Hypertext Markup Language (HTML)*

HTML atau *Hypertext Markup Language* merupakan bahasa yang paling umum digunakan dalam pembangunan web. Sesuai dengan namanya, markup language yang berarti bahasa penandaan, HTML digunakan untuk menandai secara khusus bagian – bagian yang terdapat dalam sebuah halaman web. Penandaan pada setiap bagian menggunakan tag (Duckett, 2010).

Pada penggunaan tag HTML tanda kurung siku “< >” untuk memulai sebuah *code* serta adanya kata atau huruf di dalamnya. Dalam sebuah struktur file HTML pada setiap bagian menandakan yang mana *header, title, body, paragraph, tabel*, dan lainnya. Untuk memulai menulis sebuah kode HTML, bisa menggunakan program aplikasi *text editor* seperti *Notepad, Sublime Text, Adobe Dreamweaver*, dan sebagainya.

2.2.6.2 *Cascading Style Sheet (CSS)*

Kode pemrograman yang bertujuan untuk menghias dan mengatur gaya tampilan/layout halaman web supaya lebih menarik dan terlihat elegan. *Cascading Style Sheet* atau lebih dikenal dengan CSS dalam pemrograman web berfungsi untuk mengatur *style* dari halaman *web* (Duckett, 2010).

Dalam penulisan kode CSS memerlukan sebuah *rule* yang dapat diatur sesuai keinginan *programmer*. *Rule* dibagi menjadi dua bagian:

1. *Selector*

Menunjuk atau memilih elemen mana yang akan diberikan *rule* CSS.

2. *Declaration*

Menetapkan bagaimana sebuah elemen yang dipilih akan ditata. *Declaration* dibagi menjadi dua bagian yang dipisahkan dengan tanda titik dua “;” yaitu:

- a. *Property*

Digunakan untuk memilih properti apa yang akan digunakan untuk memberikan efek kepada elemen yang telah dipilih.

- b. *Value*

Memberikan nilai secara spesifik bagi *property* yang telah dipilih

Oleh karena itu jika file HTML tidak memiliki *rule* CSS, maka tampilan

yang diberikan HTML akan tampak tidak rapi atau kurang beraturan. Jika diterapkan *rule* CSS pada *elemen* atau *tag* pada file HTML maka tampilan yang diberikan menjadi lebih baik.

2.2.6.3 JavaScript

JavaScript merupakan bahasa pemrograman yang digunakan dalam pemrograman web (Duckett, 2010).

Penulisan *code* JavaScript ditandai dengan *tag* ‘<script type="JavaScript"> </script>’ pada halaman *HTML*. *Code* JavaScript dapat dimasukkan secara internal ataupun eksternal.

Penulisan secara internal cukup dengan memasukan *code* di antara *tag script*. Sedangkan secara eksternal, *code* berupa file yang disimpan berlainan folder ataupun dari *link* internet dituliskan seperti berikut: <script type = "JavaScript" src ="scripts/jquery.js"> </script>.

Hal yang didapat jika menggunakan *JavaScript*:

1. Membaca dan menuliskan (*read & write*) elemen dan teks.
2. Memanipulasi atau memindahkan teks.
3. Menjalankan fungsi perhitungan pada data.
4. Bereaksi terhadap *events*, seperti saat pengguna menekan sebuah *button*.
5. Membaca waktu dan tanggal di komputer pengguna.
6. Menetapkan ukuran tampilan beserta resolusi layar dan web browser *version* pada komputer pengguna.
7. Melakukan pengecekan pada *input* yang diberikan oleh pengguna, seperti pada pengecekan *form validation*.

2.2.6.4 PHP

PHP (*Hypertext Processor*) adalah bahasa pemrograman yang memungkinkan para *web developer* untuk membuat aplikasi *web* yang dinamis dengan cepat dan mudah. PHP dirintis dan diperkenalkan pertama kali sekitar tahun 1994 oleh Rasmus Lerdorf melalui situsnya untuk mengetahui siapa saja yang telah mengakses ringkasan *online*-nya (Gunawan, 2010).

PHP disebut bahasa pemrograman *server-side* karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman

client-side seperti JavaScript yang diproses pada web *browser* (client).

PHP dirancang untuk membentuk web dinamis, gunanya membentuk suatu tampilan berdasarkan permintaan terkini. Untuk sebuah web jika menggunakan bahasa pemrograman PHP maka akan menjadi lebih dinamis dan rapi. kode PHP biasanya di sisipkan kedalam file HTML, karena itu PHP disebut juga sebagai *Scripting Language* atau bahasa pemrograman *script*.

PHP memiliki kelebihan dibanding dengan bahasa *scripting* lainnya. Berikut ini kelebihannya (Welling & Thompson, 2009):

1. *Performance*: PHP dapat bekerja cepat dengan performa yang sangat baik.
2. *Scalability*: Skala komoditas yang besar.
3. Integrasi ke *database*: PHP memiliki *native connection* yang tersedia untuk berbagai macam *database system*.
4. *Built-in libraries*: PHP memiliki banyak fungsi *built-in* untuk melakukan tugas yang berhubungan dengan *web task*.
5. *Low cost*: PHP merupakan bahasa *scripting open source*, sehingga siapapun dapat menggunakannya secara gratis.
6. *Ease of learning and use*: PHP mudah untuk dipelajari. Bahasa pemrograman PHP didasari pada bahasa pemrograman pada umumnya, seperti C, C++, Perl, dan Java.
7. *Strong object-oriented (OOP) support*: PHP versi 5 sudah didukung dengan fitur OOP.
8. *Flexibility of development approach*: PHP memberikan kemudahan bagi *developer* dalam implementasi. PHP sangat mudah untuk beradaptasi dengan *framework* seperti CodeIgniter.
9. *Availability of source code*: *Developer* memiliki akses langsung dengan *source code* PHP.
10. *Availability of support and documentation*: PHP memberikan dokumentasi yang jelas dan tersedia komunitas yang siap membantu jika *developer* mengalami kesulitan dalam menggunakan PHP.

2.2.6.5 Bootstrap

Bootstrap merupakan framework untuk membangun desain web secara responsif. Artinya, tampilan web yang dibuat oleh bootstrap akan menyesuaikan

ukuran layer dan browser yang kita gunakan baik di desktop, tablet ataupun mobile device. Dengan bootstrap kita juga bisa membangun web dinamis ataupun statis (Alatas, 2013).

2.2.6.6 XAMPP

XAMPP adalah paket program *web* lengkap yang dapat dipakai untuk belajar pemrograman *web*, khususnya PHP dan MySQL (Nugroho, 2013).

XAMPP adalah perangkat lunak bebas yang mendukung banyak sistem operasi. XAMPP merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*) yang terdiri dari program Apache HTTP Server, MySQL *database* dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X(empat sistem operasi apapun), Apache, MySQL, PHP, dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas. XAMPP merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.

Bagian penting XAMPP yang biasa digunakan pada umumnya:

1. XAMPP *Control Panel Application* berfungsi mengelola layanan (*services*) XAMPP, seperti mengaktifkan layanan (*start*) dan menghentikan layanan (*stop*).
2. Htdocs adalah *folder* tempat meletakkan berkas-berkas yang akan dijalankan.
3. PHPMyAdmin merupakan bagian untuk mengelola *database*.

2.2.6.7 MySQL

MySQL adalah sebuah *relational database server* yang sangat cepat dan dapat menyesuaikan dengan kebutuhan pengguna. MySQL merupakan sebuah basis data yang dapat menyimpan, mencari, melakukan pengurutan, dan mengambil data (Welling & Thompson, 2009).

Server MySQL mengontrol akses data sehingga dapat diakses oleh banyak pengguna secara bersamaan, menyediakan akses yang cepat, dan memastikan hanya pengguna yang terotorisasi yang dapat mengakses basis data. MySQL menggunakan SQL (*Standard Library Language*).

MySQL dan PHP merupakan sistem yang saling terintegrasi. Maksudnya

adalah pembuatan database dengan menggunakan sintak PHP dapat di buat. Sedangkan input yang di masukkan melalui aplikasi web yang menggunakan script serverside seperti PHP dapat langsung dimasukkan ke database MySQL yang ada di server dan tentunya web tersebut berada di sebuah web server.

2.2.6.8 PHPMyAdmin

PhpMyAdmin adalah sebuah aplikasi open source yang berfungsi untuk memudahkan manajemen MySQL. Dengan menggunakan phpMyAdmin, user dapat membuat database, membuat tabel, memasukkan, menghapus dan mengupdate data dengan GUI dan terasa lebih mudah tanpa perlu mengetikkan perintah SQL secara manual.

Beberapa fitur dalam phpMyAdmin :

1. Antarmuka berbasis web
2. Impor data dari CSV dan SQL
3. Ekspor data dari berbagai format
4. Membuat grafik PDF dari tampilan dan basis data user
5. Membuat kompleks query menggunakan *Query-by-example* (QBE)
6. Transformasi data disimpan ke dalam format yang menggunakan satu set yang telah ditetapkan seperti menampilkan data blob-data atau download-link.

2.2.6.9 Basis Data

Basis Data adalah kumpulan data (elementer) yang secara logika berkaitan dalam merepresentasikan fenomena/fakta secara terstruktur dalam domain tertentu untuk mendukung aplikasi pada sistem tertentu. Basis data adalah kumpulan data yang saling berhubungan yang merefleksikan fakta-fakta yang terdapat di organisasi (Hariyanto, 2004).

Database atau basis data adalah kumpulan data yang disimpan secara sistematis di dalam komputer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur, dan juga batasan-batasan data yang akan disimpan. Basis data merupakan aspek yang sangat penting dalam sistem informasi dimana basis data merupakan gudang penyimpanan data yang

akan diolah lebih lanjut. Basis data menjadi penting karena dapat menghindari duplikasi data, hubungan antar data yang tidak jelas, organisasi data, dan juga update yang rumit.

Proses memasukkan dan mengambil data ke dan dari media penyimpanan data memerlukan perangkat lunak yang disebut dengan sistem manajemen basis data (*Database Management System / DBMS*). Tujuan utama dari DBMS adalah untuk memberikan tinjauan abstrak data kepada user (pengguna). Jadi sistem menyembunyikan informasi tentang bagaimana data disimpan, dipelihara, dan tetap dapat diambil (akses) secara efisien.

2.2.6.10 SQL Injection

SQL injection merupakan salah satu teknik hacking dengan cara memanipulasi sintaks SQL. Pada sql injection dilakukan dengan memasukan perintah – perintah yang digunakan dalam database melalui *url*. Pada dasarnya *SQL Injection* merupakan cara mengeksploitasi celah keamanan yang muncul pada level atau “layer” *database* dan aplikasinya. Celah keamanan tersebut ditunjukkan pada saat penyerang memasukkan nilai “string” dan karakter-karakter contoh lainnya yang ada dalam instruksi SQL. Dikatakan sebagai sebuah “injeksi” karena aktivitas penyerangan dilakukan dengan cara “memasukkan” string (kumpulan karakter) khusus untuk melewati filter logika hak akses pada website atau sistem komputer yang dimaksud (Zam, 2015).

2.2.7 Metode Pengujian Sistem

2.2.7.1 Black Box Testing

Black box Testing atau pengujian *black box* merupakan metode uji coba yang memfokuskan pada keperluan fungsional dari software. Karna itu uji coba blackbox memungkinkan pengembang software untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program.

Black box testing berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineers* untuk memperoleh set kondisi *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program (Pressman, Pendekatan Praktisi Rekayasa Perangkat Lunak Edisi 7, 2010).

Black box testing berfokus pada spesifikasi fungsional dari perangkat lunak, *tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program.

Black box testing berusaha untuk menemukan kesalahan dalam kategori berikut :

1. Fungsi yang tidak benar atau fungsi yang hilang
2. Kesalahan antarmuka
3. Kesalahan dalam struktur data atau akses *database* eksternal
4. Kesalahan perilaku (*behavior*) atau kesalahan kinerja
5. Inisialisasi dan pemutusan kesalahan

Kategori error yang akan diketahui melalui *black box testing* :

1. Fungsi yang hilang atau tak benar
2. Error dari antar-muka
3. Error dari struktur data atau akses eksternal database
4. Error dari kinerja atau tingkah laku
5. Error dari inisialisasi dan terminasi

2.2.7.2 User Acceptance Test (UAT)

Pengujian UAT atau Uji Penerimaan Pengguna adalah suatu proses pengujian oleh pengguna yang dimaksudkan untuk menghasilkan dokumen yang dijadikan bukti bahwa software yang telah dikembangkan telah dapat diterima oleh pengguna, apabila hasil pengujian (*testing*) sudah bisa dianggap memenuhi kebutuhan dari pengguna.

User Acceptance Testing merupakan pengujian yang dilakukan oleh *end-user* dimana *user* tersebut adalah *staff* / karyawan perusahaan yang langsung berinteraksi dengan sistem dan dilakukan verifikasi apakah fungsi yang ada telah berjalan sesuai dengan kebutuhan/fungsinya (Perry, 2006).

Maka dari definisi tersebut, dapat dikatakan bahwa UAT merupakan pengujian yang dilakukan oleh pengguna dari sistem tersebut untuk memastikan fungsi-fungsi yang ada pada sistem tersebut telah berjalan dengan baik dan sesuai dengan kebutuhan pengguna. Proses dalam UAT adalah pemeriksaan dan pengujian terhadap hasil pekerjaan. Diperiksa apakah item-item yang ada dalam dokumen requirement sudah ada dalam software yang diuji atau tidak. Diuji

apakah semua item yang telah ada telah dapat memenuhi kebutuhan penggunanya.

Dapat dilihat pada tabel 2.6 adalah kriteria interpretasi skor dalam pengujian UAT .

Tabel 2. 6 Kriteria Interpretasi Skor

Presentase	Keterangan
0% - 20%	Sangat Kurang Baik
21% - 41%	Kurang Baik
41% - 60%	Cukup Baik
61% - 80%	Baik
81% - 100%	Sangat Baik

Sumber: (Riduwan, et al., 2019)