**BAB II**

**LANDASAN TEORI**

**2.1 PERANCANGAN**

Perancangan atau desain didefinisikan sebagai proses aplikasi berbagai teknik dan prinsip bagi tujuan pendefinisian suatu perangkat, suatu proses atau sistem dalam detail yang memadai untuk memungkinkan realisasi fisiknya Untuk mengendalikan proses desain, didalam jurnalnya A. Davis mengusulkan serangkaian prinsip-prinsip dasar dalam perancangan/desain sebagai berikut:

1. Desain tidak boleh menderita karena tunnel vision (visi terowongan).
2. Desain tidak boleh berulang.
3. Desain harus terstruktur untuk mengakomodasi perubahan.
4. Desain harus terstruktur untuk berdegradasi dengan baik, bahkan pada saat data dan event-event (kejadian-kejadian) menyimpang atau menghadapi kondisi operasi.
5. Desain harus dinilai kualitasnya pada saat desain dibuat, bahkan setelah jadi.
6. Desain harus dikaji untuk meminimalkan kesalahan-kesalahan konseptual (semantik).

Menurut Sucipto (2011 : 144) di dalam jurnal sisfotek global mengartikan: “Perancangan system adalah tahapan setelah analisis dari siklus pengembangan system. Dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi”.

Menurut Jurnal Networking and Security (2015 : 16) mendefenisikan: “Perancangan adalah suatu proses pemilihan dan pemikiran yang menghubungkan faktra-fakta berdasarkan asumsi-asumsi yang berkaitan dengan masa datang dengan menggambarkan dan merumuskan kegiatan-kegiatan tertentu yang diyakini diperlukan untuk mencapai tujuan-tujuan tertentu dan menguraikan bagaimana pencapaiannya”.

Berdasarkan dari beberapa defenisi diatas maka dapat disimpulkan bahwa pengertian dari perancangan adalah tahap untuk memecahkan suatu masalah dan mengembangkan solusi di dalam fase pengembangan rekayasa suatu produk atau sistem perangkat lunak.

**2.2 APLIKASI**

Sebelum aplikasi pertama di dunia ditemukan, ada sejarah yang sangat panjang dalam proses ditemukannya aplikasi. Jauh sebelum aplikasiyang ada pada computer dapat berjalan secanggih saat ini, semua perangkat lunak *(software)* selalu mengandalkan aljabar Boolean.

Aplikasi dikembangkan menggunakan bahsa pemrograman, seperti C, C++, Java, PHP, Python dan banyak lagi. Pada dasarnya, aplikasi dibangun untuk memproses instruksi dari pengguna sebagai input dan output hasil yang diinginkan sebagai output.

Menurut Rachmad Hakim S di dalam jurnal computech & bisnis (2014 : 63) “Aplikasi adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur Windows, permainan (game), dan sebagainya”.

Menurut Kamus Besar Bahasa Indonesia (2015 : 2) mendefenisikan “Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu”.

Dari beberapa pendapat tersebut dapat disimpulkan bahwa aplikasi merupakan sekumpulan elemen yang saling berinteraksi dan saling berkaitan antara satu dengan yang lainnya dalam melakukan suatu kegiatan secara bersama-sama untuk mencapai suatu tujuan tertentu.

**2.3 KONSEP PENGOLAHAN DATA**

**2.3.1 Data**

Pada dasarnya data berhubungan dengan berbagai aspek kehidupan manusia. Namun, penggunaan istilah data lebih banyak dipakai di bidang teknologi computer dan juga kegiatan penelitian.

Mengacu pada defenisi data, adapun fungsi data adalah sebagai berikut :

1. Data dapat berfungsi sebagai acuan dalam mengambil suatu keputusan dalam pemecahan masalah.
2. Data bisa dijadikan sebagai pedoman atau dasar suatu penelitian atau perencanaan.
3. Data dapat berfungsi sebagai acuan dalam implementasi suatu kegiatan.

Menurut Riky Nurjaman (2013 : 14) “Data adalah fakta baik dalam bentuk angka-angka, huruf-huruf atau apapun yang dapat digunakan sebagai input dalam proses untuk menghasilkan informasi”.

Menurut Gordon B. Davis dalam buku Jeperson Hutahaean (2014 : 8) “Data adalah bahan mentah bagi informasi, dirumuskan sebagai kelompok lambang-lambang tidak acak menunjukkan jumlah-jumlah, tindakan-tindakan, hal-hal dan sebagainya”.

Dari pengertian diatas maka dapat disimpulkan data adalah catatan atas kumpulan fakta yang masih bersifat mentah.

**2.3.2 Pengolahan Data**

Proses pengolahan data yang umum digunakan pada komputer yaitu metode pengolahan data terkumpul (*Batch Proessing*) dan metode pengolahan transaksi, atau disebut juga pengolahan *online processing.* Metode pengolahan data jenis ini banyak digunakan pada komputer yang sudah lama, saat ini metode ini sudah sangat jarang sekali ditemukan. *Batch* berarti mengumpulkan data dalam periode yang tertentu.

Menurut Al-Bahra Bin Ladjamudin (2013 : 9) di dalam jurnal sisfotek global “Pengolahan data (*Data Processing*) adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan”.

Menurut George R. Terry Ph. D dalam buku Jeperson Hutahaean (2014 : 8) “Pengolahan data adalah serangkaian operasi atas informasi yang direncanakan guna mencapai tujuan atau hasil yang diinginkan”.

Pengolahan data terdiri dari kegiatan-kegiatan sebagai berikut :

1. Penyimpanan data (*data storage*)

Penyimpanan data meliputi pekerja pengumpulan (*filling*), pencarian (*searching*), dan pemeliharaan (*maintenance*).

1. Penanganan data (*data handling*)

Penanganan data meliputi bagian kegiatan, seperti pemeriksaan (*veryfying*), perbandingan (*comparing*), pemilihan (*sorting*), peringkasan (*extracting*), dan penggunaan (*manipulating*).

Secara garis besar dapat disimpulkan bahwa pengolahan data adalah suatu pengolahan data yang dapat memanipulasi data menjadi bentuk yang lebih berarti dan berguna bagi pemakai yang berupa informasi.

**2.4 SEWA**

Kata “sewa” sebenarnya sama dengan istilah “kontrak”. Kata “sewa” banyak dipakai secara umum merujuk pada tindakan seseorang memakai barang milik orang lain untuk periode tertentu dengan membayar sejumlah uang.

Menurut Kamus Besar Bahasa Indonesia “Sewa adalah pemakaian sesuatu dengan membayar uang, uang yang dibayarkan karena memkai atau meminjam sesuatu”.

**2.5 KOS**

Mendengar kata “kost”, setiap orang hampir mempunyai kesamaan pengertian, yaitu suatu tempat atau sebuah kamar yang disewakan kepada orang yang membutuhkan dalam jangka waktu tertentu dan dengan biaya sewa tertentu.

Kos atau kos-kosan sekarang telah menjadi salah satu lading bisnis yang banyak diminati. Mulai dari kategori bisnis sampingan sampai kategori bisnis yang serius. Dikatakan sebagai bisnis sampingan, karena hanya memanfaatkan ruangan-ruangan dalam rumah atau kamar-kamar yang kosong dari pada tidak mempunyai nilai apapun. Dikatakan sebagai bisnis yang serius, karena tujuannya memang berbisnis dalam bidang penyedia tempat kost. Untuk kategori ini memang mebutuhkan modal yang besar dan usaha ini dijalankan secara professional.

Menurut Nurdini (2012 : 11) dalam jurnal nya “Rumah kos atau hunian sewa atau hunian transien merupakan hunian yang oleh pemiliknya disewakan seluruh atau sebagian unitnya kepada mahasiswa sebagai pihak pengguna untuk mewadahi kegiatan-kegiatan mahasiswa selama masa studinya dan berkesempatan dalam pengembangan dan peningkatan kualitas pendidikan di institusi akademik”.

Menurut Jhon Modell (2010 : 13) “Rumah kos adalah rumah kos yang ditempati oleh masyarakat umum yang tidak memiliki rumah pribadi dan menginginkan berdekatan dengan lokasi beraktifitas”.

Berdasarkan dari beberapa pendapat diatas dapat disimpulkan bahwa *kos-kosan* adalah rumah yang digunakan orang untuk menginap selama 1 hari atau lebih, dan kadang-kadang untuk periode waktu yang lebih lama misalnya, minggu, bulan atau tahunan.

**2.6 WEBSITE**

World wide web (WWW), lebih dikenal dengan web, merupakan salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke Intenet. Web pada awalnya adalah ruang informasi dalam Internet, dengan menggunakan teknologi hyperteks, pemakai dituntun untuk menemukan informasi dengan mengikuti link yang disediakan dalam dokumen web yang ditampilkan dalam browser web.

Tahun 1993, Tim Berners-Lee dan peneliti lain di European Particle Physics Lab ( Consei Europeen pour la Recherche Nucleaire, atau CERN) diGeneva, Swiss, mengembangkan suatu cara untuk men-share data antarkoleganya menggunakan dokume pada layar kompurer dengan menggunakan software browser baru.

Dalam sebuah literatur baru-baru ini, aplikasi Web diartikan sebagai program aplikasi yang berjalan pada Internet atau Intranet, dan Ekstranet perusahaan. Pengguna aplikasi Web menggunakan browser Web pada komputer client untuk menjalankan program pada sisi server. Seluruh pemrosesan yang dikerjakan pada server akan sama seperti yang dikerjakan pada mesin loka pengguna. Ada tiga jenis aplikasi Web, yaitu:

1. Dokumen Web statis
2. Aplikasi Web interaktif sederhana
3. Sistem basis data berbasis Web yang kompleks

Aplikasi Web juga berdasarkan pada halaman Web perorangan, baik itu aplikasi statis ataupun dinamis. Hal ini memungkinkan aplikasi untuk bisa dibagi ke dalam bagian yang dianggap jelas, dan mengizinkan atau menolak akses yang dibutuhkan.

Menurut Sibero (2013 : 11) mengatakan bahwa “*Web* *server* adalah sebuah computer yang terdiri dari perangkat keras dan perangkat lunak. Secara bentuk fisik dan cara kerjanya, perangkat keras *web server* tidak berbeda dengan computer rumah atau PC, yang membedakannya adalah kapasitas dan kapabilitasnya ”.

Menurut Buku Rekayasa Web (2010 : 47) “Web adalah sebuah sistem dengan informasi yang disajikan dalam bentuk teks, gambar, suara, dan lain-lain yang tersimpan dalam sebuah server web internet yang disajikan dalam bentuk hypertext. Informasi web dalam bentuk teks umumnya ditulis dalam format HTML (*Hypertext Markup Language*)”.

Berdasarkan dari beberapa pendapat diatas dapat disimpulkan bahwa *web* adalah halaman-halaman dalam internet *webserver* yang mengandung banyak informasi yang diinginkan dengan cara mengikuti *link* (*hyperlink*) yang disediakan didalam sebuah dokumen yang ditampilkan oleh aplikasi *web browser.*

**2.7 ALAT BANTU PEMODELAN SISTEM**

**2.7.1 UML (*Unified Modeling Language*)**

Notasi UML dibuat sebagai kolaborasi dari Grady Booch, DR.James Rumbough, Ivar Jacobson, Rebecca Wirfs-Brock, Peter Yourdon, and lainnya. Jacobson menulis tentang pendefenisian persyaratan-persyaratan sistem yang disebut *use case*. Juga mengembangkan sebuah metode untuk perancangan sistem yang disebut *Object-Orioented Software Enginnering (OOSE)* yang berfokus pada analisis. Booch, Rumbough dan Jacobson biasa disebut dengan tiga sekawan *(tree amigos*).Semuanya bekerja di *Rational Software Corporation* dan berfokus pada standarisasi dan perbaikan ulang UML. Simbol UML mirip dengan Booch, notasi OMT, dan juga ada kemiripan dengan notasi lainnya.

Menurut Nugroho (2009 : 4) “UML (*Unified Modeling Language*) adalah metodologi kolaborasi antara metoda-metoda Booch, OMT (*Object Modeling Technique*), serta OOSE (Object Oriented Software Enggineering) dan beberapa metoda lainnya, merupakan metodologi yang paling sering digunakan saat ini untuk analisa dan perancangan sistem dengan metodologi berorientasi objek mengadaptasi maraknya penggunaan bahasa “pemrograman berorientasi objek” (OOP)”.

Menurut buku Unified Modeling Language (2018 : 49) “UML adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi objek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku”.

Berdasarkan pendefenisian dari beberapa menurut istilah diatas dapat disimpulkan bahwa *UML* merupakan kesatuan dari bahasa pemodelan yang dikembangkan oleh Booch, Object Modeling Technique (OMT) dan Object Oriented Software Engineering (OOSE).

**2.7.1.1 Use Case Diagram**

*Use case* atau *diagram use case* merupakan pemodelan untuk melakukan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan use case.

1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2. Use case meurupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling betukar pesan antarunit atau aktor.

Menurut Buku Rekayasa Perangkat Lunak Menggunakan UML dan Java (2009 : 7) “*Use case diagram* merupakan deskripsi lengkap tentang interaksi yang terjadi antara para actor dengan sistem/perangkat lunak yang sedang kita kembangkan”.

Menurut Jurnal Algoritma Sekolah Tinggi Teknologi Garut (2012 : 6) “Use case adalah konstruksi untuk mendeskripsikan bagaimana sistem terlihat dimata pengguna”.

Berdasarkan pendefenisian dari beberapa menurut istilah diatas dapat disimpulkan bahwa *use case* merupakan suatu sistem yang berupa komponen-komponen yang terjadi antara para *aktor* dan sistem yang dapat dikembangkan.

**Tabel 2.1 Simbol *Use Case***

(Rosa A.S dan M.Shalahuddin : 2011)

|  |  |
| --- | --- |
| **SIMBOL** | **DESKRIPSI** |
| Use case | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama use case. |
| Aktor / actor  **nama aktor** | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor. |
| Asosiasi/association | Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor. |
| Ekstensi/extend  <<extend>>  .................................. | Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu, mirip dengan prinsip inheritance pada pemrograman berorientasi objek, biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan misal  <<extend>>    <<extend>>  Arah panah mengarah pada *use case* yang ditambahkan. |
| Generalisasi/generalization | Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya:  Arah panah mengarah pada *use case* yang menjadi generalisasinya (umum). |
| Menggunakan/include/uses  <<include>>  ……………………….  ‹‹uses›› | Relasi *use case* tambahan ke seluruh *use case* dimana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini.  Ada dua sudut pandang yang cukup besar mengenai include di *use case*:   * Include berarti *use case* yang ditambahkan akan selalu dipanggil saat *use case* tambahan dijalankan, misal pada kasus berikut:   <<include>>   * Include berarti *use case* yang tambahan akan selalu melakukan pengecekan apakah *use case* yang ditambahkan telah dijalankan sebelum use case tambahan dijalankan, missal pada kasus berikut:   <<include>>    Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan. |

**2.7.1.2 Activity Diagram**

Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokkan tampilan dari sistem/user interface di mana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.

Menurut Andri Anto Tri Susilo (2014 : 12) di dalam jurnalnya mendefenisikan “Diagram activity merupakan bagian dari penggambaran system secara fungsional menjelaskan proses-proses logika atau fungsi yang terimplementasi oleh kode program”.

Menurut Rizqi Fitriansyah Antasari (2012 : 39) di dalam jurnalnya “Activity diagram adalah gambaran aliran kejadian suatu aplikasi. Activity diagram menggambarkan proses yang berjalan di system yang di lakukan oleh pengguna”.

Jadi dapat disimpulkan bahwa *Activity Diagram* merupakan suatu aktivitas yang menggambarkan proses atau urutan aktivitas yang sedang dirancang. *Activity Diagram* juga dapat memoddelkan aliran kerja *(workflow)* dari *use case* bisnis dalam bentuk grafik. Ini snagat bermanfaat dalam memodelkan sebuah proses untuk membantu dalam memahami proses tersebut secara keseluruhan.

**Tabel 2.2 Simbol *Activity Diagram***

(Rosa A.S dan M.Shalahuddin : 2011)

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| Status awal | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal. |
| Aktivitas  aktivitas | Aktivitas yang dilakukan system, aktivitas biasanya diawali dengan kata kerja. |
| Percabangan/decision | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu. |
| Penggabungan/join | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
| Status akhir | Status akhir yang dilakukan system, sebuah diagram aktivitas memiliki sebuah status akhir. |
| Swimlane  nama swimlane  Atau  Nama swimlane | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |

**2.7.1.3 Class Diagram**

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

1. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Susunan struktur kelas yang baikm pada diagram kelas sebaiknya memiliki jeis-jenis kelas berikut:

1. Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

1. Kelas yang menangani tampilan sistem

Kelas yang mendefinisikan dan mengatur tampilan ke pemakai

1. Kelas yang diambil dari pendefinisian use case

Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian use case.

1. Kelas yang diambil dari pendefinisian data

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Menurut Andri Anto Tri Susilo ( 2014 : 12) “*Class diagram* merupakan diagram yang selalu ada di permodelan system berorientasi objek”.

Menurut Rosa A.S & M.Shalahuddin (2013 : 141) “Diagram kelas atau *class diagram* menggambarkan struktur system dari segi pendefenisian kelas-kelas yang akan dibuat untuk membangun system. Kelas memiliki apa yang disebut atribut dan metode atau operasi”.

Jadi dapat disimpulkan bahwa *Class Diagram* adalah gambaran dari struktur sebuah sistem atau alat perancangan untuk pengembangan sebuah sistem.

**Tabel 2.3 Simbol *Class Diagram***

(Rosa A.S dan M.Shalahuddin : 2011)

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| Kelas   |  | | --- | | **nama\_kelas** | | +atribut | | +operasi() | | Kelas pada struktur sistem. |
| Antarmuka/interface    **nama\_interface** | Sama dengan konsep *interface* dalam pemrograman berorientasi objek. |
| Asosiasi/association | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity.* |
| Asosiasi berarah/directed association | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity.* |
| Generalisasi | Relasi antar kelas dengan makna generalisasi-dpesialisasi (umum khusus). |
| Kebergantungan/dependency  ………………………… | Relasi antar kelas dengan makna kebergantungan antar kelas. |
| Agregasi/aggregation | Relasi antar kelas dengan makna. |

**2.8 ALAT BANTU PENGEMBANGAN PROGRAM**

**2.8.1 PHP**

Php dikatakan sebagai sebuah *server-side* *embedded script language* artinya *syntax-syntax* dan perintah yang kita berikan akan sepenuhnya dijalankan oleh server tetapi disertakan pada halaman HTML biasa. Aplikasi-aplikasi yang dibangun oleh PHP pada umumnya akan memberikan hasil pada *web browser*, tetapi prosesnya secara keseluruhan dijalankan di server.

Menurut Adelheid dan Nist (2012 : 2) “PHP atau *Hypertext Processor* merupakan bahasa berbentuk script yang ditempatkan dalam server dan dieksekusi didalam server untuk selanjutnya ditransfer dan dibaca oleh Client. Php juga bisa disisipkan dalam bahasa HTML”.

­Menurut tim EMS (2012 : 61) “PHP adalah bahasa pelengkap HTML yang memungkinkan dibuatnya aplikasi dinamis yang memungkinkan adanya pengolahan data dan pemrosesan data”.

Jadi dapat disimpulkan bahwa pengertian dari PHP (*Hypertext Processor*) adalah bahasa pemrograman yang dirancang untuk membangun sebuah aplikasi *web*, yang mempunyai fungsi fisik.

Selain PHP, sebenarnya ada beberapa alternative teknologi sejenis. Masing-masing teknologi memiliki kelebihan dan kekurangan. Beberapa teknologi sejenis yang cukup popular diantaranya adalah :

1. Active server pages (ASP)

ASPmerupakan produk komersial yang dikembangkan Microsoft. Teknologi ini menggunakan basis Microsoft IIS *(internet information service).*

1. Cold Fusion Markup Language (CFML)

CFML merupakan produk komersial yang dikembangkan macromedia. Teknologi ini menggunakan *cold fusion server.*

1. Java Server Pages (JSP)

JSP merupakan teknologi yang didukung oleh Sun Micro Systems. Ini dikembangkan dari bahasa pemrograman Java, yang dikenal dengan portabilitas dan kompatibilitasnya.

**2.8.2 MySQL**

MySQL dikembangkan oleh sebuah perusahaan swedia bernama MySQL AB yang pada saat itu bernama TcX DataKonsult AB sekitar Tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak 1979. Awalnya membuat MySQL dengan tujuan mengembangkan aplikasi *web* untuk klien. TcX merupakan perusahaan pengembang software dan konsultan database. Saat ini MySQL sudah diakuisi oleh Oracle Corp.

Menurut Riky Nurjaman (2013 : 18) mendefenisikan “MySQL adalah sebuah perangkat lunak system manajemen basis data *SQL* atau dalam bahasa inggris *database management system (DBMS)* yang *multi-user,* dengan sekitar 6 juta instansi di seluruh dunia”.

Menurut Yudhanto dan Purbayu (2014 : 2) “MySQL adalah salah satu system manajemen data base *(Data Base Management System)* atau biasa disebut dengan DBMS, yang terpisah, disamping karena kehandalannya, kecepatan,dan fleksibilitasnya”.

Sedangkan menurut Kurniawan (2010 : 16) “MySQL merupakan suatu jenis database server yang sangat terkenal. MySQL termasuk jenis RDBMS (Relational Database Manajement Sistem)”.

Dari beberapa defenisi diatas dapat disimpulkan MySQL adalah salah satu jenis database yang digunakan untuk membangun sebuah aplikasi *web* yang menggunakan *database server* yang mampu menerima dan mengirim data dengan cepat.

**2.8.3 XAMPP**

XAMPP dikembangkan dari sebuah tim proyek bernama Apache Friends, yang terdiri dari Tim Inti (*Core Team*). Tim Pengembang (*Development Team)* dan Tim Dukungan (*Support Team).*

XAMPP adalah singkatan yang masing-masing hurufnya adalah :

1. X, Program ini dapat dijalankan dibanyak sistem operasi, seperti Windows, Linux, Mac OS, dan Solaris.
2. A, Apache, merupakan aplikasi web server
3. M, MySQL, merupakan aplikasi database server.
4. P, PHP, bahasa pemrograman web.
5. P, perl, bahasa pemrograman untuk segala keperluan.

Menurut Srilestanti (2016 : 72) didalam jurnalnya “XAMPP merupakan tool yang menyediakan paket perangkat lunak kedalam satu buah paket”.

Menurut Aditya (2011 : 16) mengatakan “XAMPP adalah perangkat lunak bebas yang mendukung banyak system operasi, merupakan kompilasi dari beberapa program : fungsinya adalah sebagai server yang berdiri sendiri *(localhost)*, yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat system operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan bebas. Merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis”.

Jadi, dari defenisi diatas dapat disimpulkan bahwa XAMPP merupakan suatu paket yang digunakan untuk pengembangan aplikasi *web.* Dengan menginstall XAMPP maka tidak perlu lagi melakukan instansi dan konfigurasi web server Apache, PHP dan MySQL secara manual.

**2.8.4 Java Script**

Javascript dikembangkan pertama kali oleh Netscape, kemudian diadopsi juga oleh Microsoft sebagai bahasa *script*  untuk *browser* di Internet Explorer, dengan nama Jscript.

Javascript dibuat agar dapat digunakan juga oleh pemrograman *web* yang bukan berlatar pemrogram computer, pemrogram computer yang tidak berlatar belakang pemrogram yang terdidik dengan pemrograman berorientasi objek. Javascript juga dirancang agar Netscape Navigator dapat mendukung aplet (aplikasi Java yang dapat dijalankan di atas *browser web*). Nama Javascript diambil berdasarkan persetujuan dari Sun Microsystem yang mengembangankan Java, diumumkan bersama oleh Netscape dan Sun pada tanggal 4 Desember 1995. Sebelumnya Javascript bernama LiveScript.

Menurut Irawan (2008 : 3) “Javascript merupakan bahasa pemrograman berbasis script. Javascript memiliki kemampuan untuk menciptakan halaman web yang dinamis serta didukung oleh banyak web browser”.

Menurut Yeni Kustiyaningsih dan Devie Rosa Anamisa (2011 : 138) “Javascript adalah bahasa yang berbentuk kumpulan script berjalan pada satu dokumen HTML. Bahasa pemrograman javascript adalah salah satu bahasa pemrograman komputer yang sangat mampu memberikan kemampuan lebih terhadap bahas html dengan diizinkannya pengeksekusian perintah perintah yang diisi oleh user. Hal ini berarti pada sisi browser bukan pada server web.”

Jadi, dari defenisi diatas dapat disimpulkan bahwa Javascript adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokumen HTML yang ditampilkan dalam browser menjadi lebih interaktif, tidak sekedar indah saja. Javascript memberikan beberapa fungsionalitas ke dalam halaman *web*, sehingga dapat menjadi sebuah program yang disajikan dengan menggunakan antarmuka *web.*

**2.8.5 Sublime Text**

Sublime text mendukung berbagai bahasa pemrograman yang mampu meyajikan fiture syntax highlight hampir disemua bahsaa pemrograman yang didukung atau dikembangkan oleh pemerintah seperti: C, C++, C#, CSS, D, Dylan, Erlang, HTML. Biasanya bagi bahasa pemrograman yang didukung ataupun yang belum terdukung secara default dapat dimaksimalkan atau didukung dengan menggunakan add-ons yang bisa didownload sesuai keubutuhan user.

Menurut Geovanne Farel (2018:58) di dalam jurnalnya mengatakan “Sublime Text adalaha aplikasi editor untuk kode dan text yang dapat berjalan diberbagai platform operating system dengan menggunakan teknologi phyton API. Terciptanya aplikasi ini terinsiprasi dari aplikasi VIM, aplikasi ini sangatlah fleksibel dan sangatlah powerfull”.

Jadi dapat disimpulkan bahwa sublime text adalah teks editor untuk berbagai bahasa pemrograman termasuk pemrograman PHP. Sublime Teks merupakan teks editor lintas platform dengan Phyton Application Interface (API). Sublime teks juga mendukung bahasa pemrograman dan bahsa markup dan fungsinya dapat ditambah dengan *plugin*.