**BAB II**

**Landasan Teori**

**2.1 Game**

Menurut K.T Martono(2015:5),Game adalah sebuah kegiatan yang melibatkan pengguna (pemain aktif maupun lawan) untuk menuju sebuah pencapaian tertentu, dimana para pengguna diberi fasilitas untuk mengelola sumberdaya yang telah disediakan dalah game untuk mencapai tujuan. Game merupakan salah satu media yang dapat digunakan dalam menyampaikan sebuah tujuan. Tujuan yang terdapat dalam game mempunyai macam-macam jenis yaitu pendidikan, hiburan dan simulasi. Dalam sejarah kehidupan manusia, game selalu ada dan terus diminati oleh berbagai kalangan di segala usia. Game sendiri sudah ada sejak beribu-ribu tahun yang lalu dalam bentuk permainan tradisional. Di berbagai negara, terdapat permainan tradisional tersendiri sesuai dengan budaya masing-masing Negara.

**2.2 Website**

Menurut Ahmad Josi(2017:9), Website adalah kumpulan dari halaman-halaman situs, yang terangkum dalam sebuah domain atau subdomain, yang tempatnya berada di dalam *World Wide Web* ( WWW ) di dalam internet. website juga dapat diartikan sebagai sebuah halaman yang berisi data, baik data text, gambar, suara dan lainya yang dapat diakses secara online.

**2.3 Visual Studio Code**

Visual Studio Code adalah aplikasi editor teks gratis di kembangkan oleh Microsoft yang dapat digunakan di semua bahasa pemrograman yang ada tanpa perlu berganti aplikasi editor, serta dapat dijalankan di berbagai platform *Operating System* (OS) seperti windows,linux ,dan mac OS. Visual Studio Code memudahkan para Programmer saat berganti bahasa pemrograman tanpa perlu berganti aplikasi editor serta memahami dan konfigurasi tools kembali di aplikasi editor barunya.Visual Studio Code juga memberikan kebebasan kepada penggunanya dalam *tema*, *debugger* ,*extension*, dan lainnya.

**2.4 HTML**

Budi Rahajoet, dkk (2011:286) didalam buku Modul Pemrograman Web (HTML, PHP & MySQL) menjelaskan bahwa HTML adalah singkatan dari *HyperText Markup Language*. HTML merupakan file teks yang ditulis menggunakan aturan-aturan kode tertentu untuk kemudian disajikan ke user melalui suatu aplikasi web browser.Dapat disimpulkan bahwa HTML (*HyperText Markup Language*) ialah bahasa pemrograman untuk membuat desain *body* atau *layout* (tampilan) dari sebuah website. Bahasa pemrograman HTML inilah yang akan dieksekusi oleh browser dan ditampilkan dijendela browser.

**2.5 CSS**

Menurut Saputra, Subagio dan Saluky (2012:6), CSS atau *Cascading Style Sheet* merupakan suatu bahasa pemrograman web yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam web sehingga tampilan web akan lebih rapi, terstruktur, dan seragam.

**2.6 PHP**

PHP adalah sebuah bahasa *scripting server-side* yang digunakan untuk mengembangkan situs web statis, situs web dinamis, dan aplikasi web. PHP adalah kependekan *dari Hypertext Pre-processor*, sebelumnya dikenal sebagai *Personal Home Pages*.

*Script* merupakan kumpulan instruksi pemrograman yang dieksekusi pada saat runtime. Bahasa scripting adalah bahasa yang mengeksekusi skrip saat *runtime* dan biasanya diintegrasikan ke dalam lingkungan perangkat lunak lain.Karena PHP berjalan di sisi server, jenis bahasa pemrograman ini memungkinkan script atau program untuk dijalankan dan diproses oleh server. Hal ini berbeda dengan JavaScript yang berjalan di sisi klien (client-side).

Berikut ini diantara keuntungan PHP :

1. Akses cepat, karena ditulis ditengah kode HTML, sehingga waktu respon programnya lebih cepat.
2. Murah, bahkan gratis tidak perlu membayar software ini untuk menggunakannya.
3. Mudah dipakai, fitur dan fungsinya lengkap, cocok dipakai untuk membuat halaman web dinamis.
4. Dapat dijalankan diberbagi sistem operasi, seperti Windows, Linux, Mac OS, dan berbagai varian Unix.
5. Dukungan teknis banyak tersedia. Bahkan banyak forum dan situs didedikasikan untuk *troubleshooting* berbagai masalah seputar PHP
6. Aman, pengunjung tidak akan bisa melihat kode PHP.
7. Mendukung banyak database.
8. Bisa dikostumisasi. Karena software ini open source.

**2.7 MYSQL**

Menurut Haris Saputro ( 2003 ) mengemukakan bahwa MySQL merupakan database server dimana pemrosesan data terjadi di server, dan client hanya mengirim data serta meminta data. Oleh karena pemrosesan terjadi di server sehingga pengaksesan data tidak terbatas. Pengaksesan dapat dilakukan dimana saja oleh siapa saja dengan catatan komputer terhubung ke server. Lain halnya dengan database dekstop dimana segala pemrosesan data seperti penambahan data atupun penghapusan data harus dilakukan pada komputer yang bersangkutan.

Ada beberapa alasan kenapa MySQL menjadi progrm database yang sangat popular dan digunakan oleh banyak orang. Alasan-alasan tersebut diantaranya ialah :

1. MySQL merupakan database yang memiliki kecepatan yang tinggi dalam melakukan pemrosesan data, dapat diandalkan dan mudah digunakan serta mudah dipelajari.
2. Koneksi, kecepatan dan keamanan membuat MySQL sangat cocok diterapkan untuk pengaksesan database, dengan menggunakan bahasa pemrogaman Perl atau PHP sebagai antarmuka.
3. MySQL dapat melakukan koneksi dengan client dengan protokol TCP/IP, *Unix socket* ( Unix ) atau *named pipes* ( NT ).
4. Multi user yaitu dalam satu database server pada MySQL dapt diakses oleh beberapa user dalam waktu yang sama tanpa mengalami konflik atau crash.
5. Security yang dimiliki database MySQL dikenal baik, karena memiliki lapisan sekuritas pada level subnetmaks, nama host dan izin akses user dengan sistem perizinan yang khusus serta pasword yang dimiliki setiap user dalam bentuk terenkripsi.
6. MySQL dapat menangani database dengan skala yang sangat besar dengan record mencapai lebih dari 50 juta, dapat menampung 60 ribu tabel, dan juga bisa menampung 5 milyar baris data.

**2.8 XAMPP**

Menurut Imam Mulhim,(2013:4).XAMPP adalah paket instalasi program yang terdiri atas program apache HTTP Server,MySQL, database dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan perl. Software ini biasa digunakan untuk pengujian aplikasi web melalui localhost. Aplikasi web yang dikembangkan secara native, menggunakan *framework*, maupun *CMS* bisa diuji menggunakan XAMPP.

XAMPP dikembangkan oleh Apache Friends. Apache Friends sendiri merupakan proyek nirlaba yang bertujuan untuk mempromosikan server web Apache. Secara umum, XAMPP berfungsi sebagai web server yang bisa diakses melalui localhost komputer. Aplikasi web yang belum sepenuhnya jadi, bisa dikembangkan dalam server lokal ini. Developer bisa menguji seluruh fungsi aplikasi web, menguji coba desain web, menyusun struktur konten, dan hal-hal lain sebelum resmi diluncurkan ke internet.

**2.9 Database**

Merupakan sekumpulan data atau informasi yang tersimpan secara sistematis. Database memiliki peran penting dalam perangkat untuk mengumpulkan informasi, data, atau file secara terintegrasi.Fungsi database adalah untuk menghindari data ganda yang tersimpan. Suatu database management system (DBMS) dapat diatur supaya bisa mengenali duplikasi data ketika diinput. Namun selain untuk menghindari data ganda, database memiliki fungsi lainnya, antara lain:

1. Mengelompokan data dan informasi. Memudahkan dalam identifikasi data.
2. Memudahkan proses akses, menyimpan, pembaharuan, dan penghapusan data.
3. Menjadi alternatif terkait masalah penyimpanan ruang dalam suatu aplikasi.
4. Menjaga kualitas data yang diakses sesuai input.
5. Menunjang kinerja aplikasi yang memerlukan penyimpanan data.

**2.10 UML**

Menurut Sri Mulyani (2016:48), UML(*Unified Modeling Languange*) adalah sebuah teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada system.

UML Merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks.UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hamper dalam semua bidang yang membutuhkan pemodelan.

Tujuan dari penggunaan UML yaitu untuk membagikan model yang akan digunakan dan termasuk dalam permodelan visual yang ekspresif sehingga dapat saling berbagi model dengan mudah dan memperluas program.

Secara umum, UML dibagi Menjadi 3 bagian yaitu :

1. *Structure Diagram*

Berikut adalah beberapa diagram yang termasuk dalam kategori structure diagram.

* *Class Diagram*, adalah jenis diagram pada UML yang digunakan untuk menggambarkan paket-paket atau kelas-kelas yang ada pada sistem dan akan digunakan pada suatu sistem.
* *Component Diagram*, merupakan suatu diagram yang dibuat untuk menunjukkan struktur dan ketergantungan antara komponen yang ada di dalam suatu sistem. Komponen tersebut adalah hal fisik dari program yang dimodelkan dan akan terlihat saat sistem dijalankan.
* *Deployment Diagram*, adalah tata letak program secara fisik atau menampilkan bagian aplikasi yang ada pada perangkat keras dan digunakan untuk menerapkan sistem.

1. *Behavior Program*

Berikut adalah beberapa diagram yang termasuk dalam kategori behaviour diagram.

* *Use case diagram*, adalah urutan interaksi yang memiliki keterkaitan antara sistem dan aktor. Use case diagram dijalankan dengan cara menggambarkan tipe interaksi yang terjadi diantara user yang terlibat di dalam sistem.
* *Activity Diagram*, merupakan diagram yang menunjukan model dari metode yang ada di dalam sistem. Diagram dapat menggambarkan perancangan sistem, proses awal sistem berjalan, keputusan yang dapat terjadi dalam sistem, dan akhir dari sistem.

1. *Interaction Diagram*

Berikut adalah diagram yang termasuk dalam kategori interaction diagram.

* *Sequence Diagram*, adalah diagram UML yang menggambarkan hubungan di objek yang di dasarkan pada urutan waktu.
* *Communication Diagram*, merupakan jenis diagram yang terdapat pada UML untuk menjelaskan proses terjadinya aktivitas dan interaksi pada suatu sistem.

**2.10.1 Use Case Diagram**

Use Case Diagram adalah satu jenis dari diagram UML (*Unified Modelling Language*) yang menggambarkan hubungan interaksi antara sistem dan aktor. Use Case dapat mendeskripsikan tipe interaksi antara si pengguna sistem dengan sistemnya. Use Case merupakan sesuatu yang mudah dipelajari. Langkah awal untuk melakukan pemodelan perlu adanya suatu diagram yang mampu menjabarkan aksi aktor dengan aksi dalam sistem itu sendiri, seperti yang terdapat pada Use Case.

|  |  |  |
| --- | --- | --- |
| Nama | Deskripsi | Simbol |
| Actor | Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case |  |
| Use Case | Abstraksi dan interaksi Antara sistem dan aktor |  |
| Relationship | Abstraksi dari penghubung Antara actor dengan use case |  |
| Menunjukan spesialisasi actor untuk dapat berpartisipasi dengan use case |  |
| Menunjukkan bahwa suatu use case seseluruhnya merupakan fungsionalitas dari use case lainya |  |
| Menunjukkan bahwa suatu use case merupakan tambahan fungsionalitas dari use case lainya jika suatu kondisi terpenuhi |  |

**2.10.2 Class Diagram**

Class diagram atau diagram kelas adalah salah satu jenis diagram struktur pada UML yang menggambarkan dengan jelas struktur serta deskripsi class, atribut, metode, dan hubungan dari setiap objek. Ia bersifat statis, dalam artian diagram kelas bukan menjelaskan apa yang terjadi jika kelas-kelasnya berhubungan, melainkan menjelaskan hubungan apa yang terjadi.Diagram kelas ini sesuai jika diimplementasikan ke proyek yang menggunakan konsep *object-oriented* karena gambaran dari class diagram cukup mudah untuk digunakan.

Menggunakan diagram kelas memberikan banyak keunggulan bagi proses pengembangan perangkat lunak. keunggulan dari diagram kelas yaitu dapat menjelaskan suatu model data untuk sebuah program, baik model data sederhana maupun kompleks serta dapat membantu pengguna dalam menyampaikan kebutuhan dari suatu sistem.

|  |  |  |
| --- | --- | --- |
| Nama | Keterangan | Simbol |
| Generalization | Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk |  |
| Nary Association | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek |  |
| Class | Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama |  |
| Collaboration | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil terukur bagi suatu aktor |  |
| Realization | Objek yang benar benar dilakukan oleh suatu objek |  |
| Dependency | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung pada elemen yang tidak mandiri |  |

**2.10.3 Component Diagram**

Component Diagram adalah diagram UML yang menampilkan komponen dalam system dan hubungan antara mereka. Saat berurusan dengan dokumentasi sistem yang kompleks, component diagram dapat membantu memecah sistem menjadi komponen yang lebih kecil. Fungsi utama dari component diagram adalah untuk memvisualisasikan bagaimana komponen-komponen sistem saling terhubung dan berinteraksi.

|  |  |  |
| --- | --- | --- |
| Package | Bungkusan dari satu atau lebih komponen |  |
| Komponen | Komponen sistem |  |
| Dependency | Ketergantungan antar komponen,arah panah mengarah pada komponen yang dipakai |  |
| Interface | Antarmuka komponen |  |

**2.10.4 Deployment Diagram**

Diagram Deployment merupakan salah satu model diagram dalam UML yang digunakan untuk menggambarkan penempatan artefak perangkat lunak pada node. Fungsinya adalah untuk mengilustrasikan hubungan antara perangkat lunak dan perangkat keras. Secara khusus, diagram Deployment dapat membantu dalam membuat model fisik tentang bagaimana komponen perangkat lunak (artefak) digunakan pada komponen perangkat keras yang disebut sebagai *node*. Diagram ini sangat penting dalam tahap implementasi perangkat lunak dan biasanya dibuat sebelum proses coding dimulai.

|  |  |  |
| --- | --- | --- |
| Package | Bungkusan dari satu atau lebih komponen |  |
| Node | merujuk pada elemen dalam diagram yang mewakili entitas fisik yang dapat menjalankan perangkat lunak atau komponen perangkat keras dalam sebuah sistem. |  |
| Dependency | Ketergantungan antar komponen,arah panah mengarah pada komponen yang dipakai |  |
| Link | Relasi antar node |  |

**2.10.5 Sequence Diagram**

Sequence diagram atau diagram urutan adalah sebuah diagram yang digunakan untuk menjelaskan dan menampilkan interaksi antar objek-objek dalam sebuah sistem secara terperinci.sequence diagaram juga akan menampilkan pesan atau perintah yang dikirim, beserta waktu pelaksanaannya. Objek-objek yang berhubungan dengan berjalannya proses operasi biasanya diurutkan dari kiri ke kanan.Tujuanya adalah untuk mengetahui urutan kejadian yang dapat menghasilkan output yang diinginkan.

|  |  |  |
| --- | --- | --- |
| Actor | Menggambarkan user yang sedang berinteraksi dengan sistem |  |
| Entity Class | Menggambarkan Hubungan yang akan dilakukan |  |
| Boundary Class | Menggambarkan sebuah gambaran dari form |  |
| Control Class | Menggambarkan penghubung Antara boundary dengan tabel |  |
| A focus of control and A life Timeline | Menggambarkan tempat mulai dan berakhirnya message |  |
| A message | Menggambarkan Pengiriman pesan |  |

**2.10.6 Activity Diagram**

Activity Diagram adalah salah satu jenis diagram pada Unified Modeling Language (UML) yang digunakan untuk menggambarkan aliran kerja atau urutan aktivitas dalam suatu sistem atau proses bisnis. Diagram ini digunakan untuk memodelkan proses secara visual dan membantu dalam memahami bagaimana aktivitas dan tugas saling terkait dalam sebuah sistem. Fungsi utama dari activity diagram adalah untuk menyajikan urutan aktivitas, keputusan, dan kontrol yang terjadi selama eksekusi suatu tugas atau proses. Diagram ini membantu dalam memahami dan menggambarkan langkah-langkah yang harus diambil untuk mencapai tujuan tertentu.

|  |  |  |
| --- | --- | --- |
| Status awal | Sebuah diagram aktivitas memiliki sebuah status awal |  |
| Aktivitas | Aktivitas yang dilakukan sistem,aktivitas biasanya diawali dengan kata kerja |  |
| Percabangan | Percabangan dimana ada pilihan aktivitas yang lebih dari satu |  |
| Penggabungan | Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan menjadi 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 |  |

**2.10.7 Communication Diagram**

Communication Diagram adalah suatu diagram yang mengambarkan struktur interaksi yang terdapat disekitar objek (seperti halnya sequence diagram) dan hubungannya terhadap yang lainnya.Communication Diagram lebih mengedepankan peran setiap objek dan bukan pada waktu pengiriman pesan (message).

Communication diagram mirip dengan sequence diagram. Yang membedakan Communication diagram dari yang lain adalah diagram yang mengarahkan objek-objek dan asosiasi dengan objek-objek yang lain dalam sistem bersamaan dengan interaksinya. Asosiasi ini tidaklah dijelaskan dalam sequence diagram.

|  |  |  |
| --- | --- | --- |
| Actor | Menggambarkan user yang sedang berinteraksi dengan sistem |  |
| Object | entitas internal dari sistem yang sedang dijelaskan atau direpresentasikan dalam diagram. |  |
| Message | representasi visual dari komunikasi yang terjadi antara objek-objek dan aktor-aktor yang terlibat dalam sistem |  |

**2.11 Sturktur Navigasi**

Struktur navigasi adalah urutan alur informasi dari suatu aplikasi multimedia. Struktur navigasi mengacu pada cara pengaturan dan penyusunan elemen-elemen interaktif di dalam suatu situs web atau aplikasi yang memungkinkan pengguna untuk berpindah dari satu halaman atau bagian ke bagian lainnya dengan mudah dan intuitif. Tujuannya adalah untuk membantu pengguna menjelajahi konten dan fitur situs web atau aplikasi dengan cepat, mudah, dan efisien.

Dalam penggambarannya Struktur Navigasi terbagi kedalam 4 Struktur yang berbeda yaitu: Linier, Non Linier, Hierarchical (Hirarki) dan Composit (Campuran).

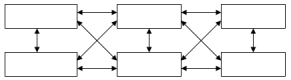
**2.11.1 Struktur Navigasi Linear**

Merupakan struktur yang hanya mempunyai satu rangkaian cerita berurut. Tampilan yang dapat ditampilkan dalam struktur ini hanya satu halaman sebelumnya atau satu halaman sesudahnya, tidak bisa lebih dari itu.

[Linear](http://hasim319.files.wordpress.com/2011/07/linear2.png)

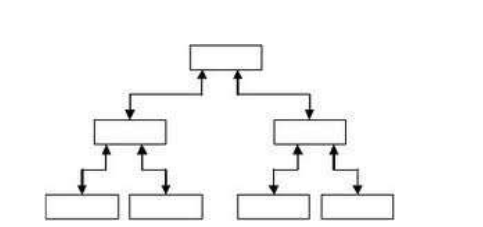
**2.11.2 Struktur Navigasi Non-Linear**

Struktur navigasi non linear ( tidak berurut ) merupakan pengembangan dari struktur navigasi linear. Pada struktur ini diperkenankan membuat navigasi bercabang. Percabangan yang dibuat pada struktur linear ini berbeda dengan percabangan pada struktur hierarki, karena pada percabangan non linear ini walaupun terdapat percabangan, tetapi tiap-tiap tampilan mempunyai kedudukan yang sama tidak ada *master page* dan *slave page*.

[](http://hasim319.files.wordpress.com/2011/07/nonlinear.png)

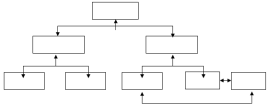
**2.11.3 Struktur Navigasi Hierarki**

Struktur navigasi hierarchi ( bercabang ) merupakan suatu struktur yang mengandalkan percabangan untuk menampilkan informasi yang berdasarkan kriteria tertentu. Informasi pada halaman utama disebut *parent* dan informasi pada cabangnya disebut *child*.



**2.11.4 Struktur Navigasi Composite**

Struktur Navigasi Composite ( campuran ) merupakan struktur gabungan dari ketiga struktur sebelumnya. Struktur ini disebut juga struktur navigasi bebas. Kelebihan dengan menggunakan struktur navigasi ini adalah suatu aplikasi mampu memberikan keterkaitan informasinya lebih baik.

[](http://hasim319.files.wordpress.com/2011/07/composite.png)