

BAB II

LANDASAN TEORI



Hak cipta milik IBI KKG (Institut Bisnis dan Informatika Kwik Kian Gie)

A. Data

Hak Cipta Dilindungi Undang-Undang

Menurut Kenneth C. Laudon dan Jane P. Laudon (2020:16), data dapat didefinisikan sebagai berikut :

”Data adalah aliran fakta mentah yang mewakili peristiwa yang terjadi dalam organisasi atau lingkungan fisik sebelum mereka telah diatur dan diatur dalam bentuk yang dapat dipahami dan digunakan oleh orang”.

Menurut J Prayoga, dkk (2023:2), Data memiliki definisi sebagai berikut :

“Data merupakan informasi yang disimpan dalam basis data. Data dapat berupa teks, angka, gambar, suara, atau jenis data lainnya. Data ini mewakili fakta atau entitas yang ingin kita simpan dan kelola”.

Data dapat dianggap sebagai fakta mentah yang belum diolah, sehingga tidak memiliki arti. Namun, data yang telah diolah akan menghasilkan konteks yang bermanfaat untuk orang lain.

B. Basis Data (*Database*)

Menurut Kenneth C. Laudon dan Jane P. Laudon (2020:216), “*Database* adalah kumpulan data yang diatur untuk melayani banyak aplikasi secara efisien dengan mensentralisasi data dan mengendalikan data yang berlebihan”.

Menurut J Prayoga, dkk (2023:1), basis data dapat didefinisikan sebagai berikut :

“*Database* adalah kumpulan terstruktur dari data yang disimpan secara elektronik. Ini adalah sistem yang digunakan untuk menyimpan, mengolah, dan mengambil informasi secara efisien”.



Database terdiri dari tabel yang terdiri dari baris dan kolom. Setiap baris mewakili satu entitas atau objek, dan setiap kolom mewakili atribut atau karakteristik dari entitas tersebut. *Database* sering digunakan dalam berbagai konteks, mulai dari bisnis dan organisasi hingga aplikasi web dan perangkat lunak

C. Sistem

Menurut Efrem G. Mallach (2020:4), “Sistem terdiri dari kumpulan komponen yang bekerja sama untuk mencapai tujuan tertentu”. Komponen ini terdiri dari subsistem yang lebih kecil, hingga partikel subatomik. Sebuah sistem memiliki batasan, yang menentukan apa yang ada di dalam dan di luarnya. Oleh karena itu, penting untuk memahami batas-batas setiap sistem.

Menurut George M. Marakas dan James O’Brien (2013:25), Sistem didefinisikan sebagai sekumpulan komponen yang saling terkait, dengan batas yang jelas, bekerja sama untuk mencapai serangkaian tujuan dengan menerima input dan menghasilkan output melalui proses transformasi yang terorganisir. Sistem memiliki tiga fungsi dasar yaitu :

1. *Input*, melibatkan pengumpulan dan pengorganisasian elemen yang masuk ke dalam sistem untuk diproses, seperti bahan mentah, energi, data, dan upaya manusia.
2. *Processing*, melibatkan transformasi input menjadi output, contohnya adalah proses manufaktur, proses pernapasan manusia, atau perhitungan matematis.
3. *Output*, melibatkan transfer elemen yang dihasilkan oleh proses transformasi ke tujuan akhirnya, seperti produk jadi, layanan manusia, dan informasi manajemen yang harus disampaikan kepada pengguna.



D. Informasi

Menurut Efrem G. Mallach (2020:6), “Informasi adalah data yang telah diatur dan diproses untuk menjadi berarti bagi seseorang yang akan menggunakannya”. Informasi juga dapat didefinisikan sebagai sesuatu yang mengurangi ketidakpastian.

Menurut Stephen Haag dan Maeve Cummings (2010:6), “Informasi adalah data yang memiliki makna tertentu dalam konteks tertentu”. Kualitas dari informasi tersebut ditentukan dari 4 atribut ini yaitu :

1. Ketepatan waktu, Informasi terdiri dari dua komponen utama. Pertama, apakah informasi dapat diakses kapanpun diperlukan? Kedua, apakah data tersebut mencerminkan rentang waktu yang relevan? Sering kali, informasi yang relevan mencakup data historis dan saat ini, yang memberikan konteks dan memungkinkan perbandingan yang diperlukan untuk pengambilan keputusan yang lebih baik.
2. Lokasi, Jika data tidak dapat diakses, data tersebut menjadi tidak berguna. Idealnya, lokasi Anda atau lokasi informasi tidak seharusnya menjadi hambatan.
3. Bentuk, Informasi memiliki dua elemen penting. Pertama, apakah informasi disajikan dalam format yang paling bermanfaat? Ini mencakup berbagai bentuk seperti audio, teks, video, animasi, atau grafis, yang dipilih berdasarkan situasi dan kemampuan pengguna. Kedua, apakah data tersebut bebas dari kesalahan? Informasi yang mengandung kesalahan dianggap kurang berkualitas dan tidak dapat digunakan secara efektif.
4. Keabsahan, Validitas adalah komponen penting kedua dalam menilai kualitas informasi. Meskipun banyak informasi tersedia di Internet, penting untuk mempertanyakan seberapa terpercaya sumbernya. Sebagian besar informasi online



tidak melalui proses kontrol atau verifikasi kualitas sebelum disebarluaskan, sehingga validitasnya sering kali diragukan.

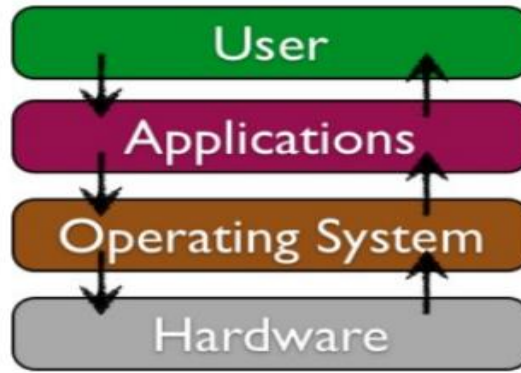
E. Sistem Informasi

Menurut David T. Bourgeois, dkk (2019:3), sistem informasi dapat didefinisikan sebagai berikut :

“Sistem informasi adalah seperangkat komponen yang saling terkait yang mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan kontrol dalam sebuah organisasi”.

Dari penjelasan David T. Bourgeois , dkk (2019:4-7), sistem informasi terdiri dari 5 komponen sebagai berikut :

1. **Hardware**, Perangkat Keras adalah bagian fisik dari sistem informasi yang dapat disentuh. *Computer, keyboard, disk drive, dan flash drive* adalah semua contoh perangkat keras sistem.
2. **Software**, Perangkat lunak terdiri dari serangkaian instruksi yang memberikan perintah kepada perangkat keras mengenai tugas yang harus dilaksanakan. Perangkat lunak bersifat non-fisik dan tidak dapat disentuh. Terdapat dua jenis perangkat lunak, yaitu :
 - a. perangkat lunak sistem operasi, berfungsi sebagai antarmuka antara perangkat keras dan perangkat lunak aplikasi
 - b. perangkat lunak aplikasi, memungkinkan pengguna untuk melakukan berbagai tugas, seperti membuat dokumen, merekam data dalam lembar kerja, atau mengirim pesan kepada orang lain.



Gambar 2. 1
Ilustrasi antara perangkat lunak dan perangkat keras

(Sumber: Bourgeois,2019:5)

© Hak cipta milik IBI KKG (Institut Bisnis dan Informatika Kwik Kian Gie)

Hak Cipta Dilindungi Undang-Undang

Institut Bisnis dan Informatika Kwik Kian Gie

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.

3. Data, Seperti halnya perangkat lunak, data juga bersifat non-fisik dan tidak dapat dilihat dalam bentuk aslinya. Potongan data yang tidak terkait mungkin tidak memiliki nilai yang signifikan. Namun, ketika data tersebut dikumpulkan, di indeks, dan diatur menjadi sebuah basis data, data tersebut dapat menjadi alat yang sangat berharga bagi bisnis.
4. *People*, Fokus pada individu yang terlibat dalam sistem informasi merupakan langkah krusial berikutnya. Mulai dari *front-line user support staff*, hingga *system analyst*, *developer*, dan *Chief Information Officer (CIO)*, setiap orang yang terlibat dengan sistem informasi memainkan peran penting. Mereka adalah elemen esensial yang memastikan sistem informasi berfungsi secara optimal dan efisien.
5. *Process*, serangkaian langkah yang diambil untuk mencapai hasil atau tujuan yang diinginkan. Sistem informasi menjadi lebih terintegrasi dengan proses organisasi, membawa produktivitas yang lebih besar dan kontrol yang lebih baik terhadap proses tersebut.



F. Web

Menurut Rosa Anamisa dan Ayu Mufarroha (2020:1), “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”.

Menurut Hassen Ben Rebah, dkk (2021:3), web dapat didefinisikan sebagai berikut :

“*World Wide Web*, biasa dikenal sebagai web, dan kadang-kadang sebagai Internet, menyajikan sistem *hypertext* yang berjalan di Internet. Web digunakan untuk memeriksa halaman yang dapat diakses di situs web menggunakan *browser*. Gambar spiderweb berasal dari *hyperlink* yang menghubungkan halaman web”.

Berdasarkan penjelasan Rosa Anamisa dan Ayu Mufarroha (2020:2), Sejarah *website* dimulai dari Sir Timothy John "Tim" Berners-Lee yang merupakan pengembang *website* pertama, dan pada tahun 1991, situs web yang pertama kali dapat terhubung ke jaringan muncul. Pada 30 April 1993, diumumkan bahwa *World Wide Web* bisa digunakan secara gratis oleh semua orang di seluruh dunia. Saat ini, *website* ditulis secara dinamis dan dikonversi menjadi *Hypertext Markup Language* (HTML), lalu diakses melalui program perangkat lunak yang dikenal sebagai *web browser*.

Dari penjelasan Ben Shneiderman, dkk (2018:95-97), delapan prinsip utama yang disebut "aturan emas" yang berlaku untuk sebagian besar sistem interaktif dan lingkungan kaya fitur. Prinsip-prinsip ini, yang dikembangkan berdasarkan pengalaman dan disempurnakan selama tiga dekade, memerlukan validasi dan penyesuaian untuk domain desain yang spesifik. Delapan Aturan Emas tersebut adalah:

1. *Strive for consistency*, Rangkaian tindakan yang konsisten harus diterapkan dalam situasi yang serupa; terminologi yang identik harus digunakan dalam *prompt*, *menu*, dan layar bantuan; serta konsistensi dalam warna, tata letak, kapitalisasi, font, dan sebagainya harus diterapkan secara keseluruhan.



2. *Seek universal usability*, Kenali kebutuhan berbagai pengguna dan desain untuk plastisitas, yang memfasilitasi transformasi konten. Perbedaan antara pemula hingga ahli, rentang usia, disabilitas, variasi internasional, dan keragaman teknologi memperkaya spektrum persyaratan yang membimbing desain. Menambahkan fitur untuk pemula, seperti penjelasan, dan fitur untuk ahli, seperti pintasan dan kecepatan yang lebih tinggi, memperkaya desain antarmuka dan meningkatkan kualitas yang dirasakan.
3. *Offer informative feedback*, Untuk setiap tindakan pengguna, harus ada umpan balik antarmuka. Untuk tindakan yang sering dan kecil, responnya dapat sederhana, sedangkan untuk tindakan yang jarang dan besar, responnya harus lebih kuat. Penyajian visual dari objek yang diminati menyediakan lingkungan yang nyaman untuk menunjukkan perubahan secara eksplisit.
4. *Design dialogs to yield closure*, Rangkaian tindakan harus diorganisasikan ke dalam kelompok dari awal, tengah, dan akhir. Umpan balik informatif pada penyelesaian kelompok tindakan memberikan kepuasan kepada pengguna, rasa lega, sinyal untuk menghapus rencana kontingensi dari pikiran mereka, dan indikator untuk mempersiapkan kelompok tindakan berikutnya.
5. *Prevent errors*, Sebisa mungkin, desain antarmuka harus mencegah pengguna dari melakukan kesalahan serius; misalnya, menonaktifkan *item menu* yang tidak sesuai dan tidak memungkinkan karakter alfabet dalam bidang entri numerik. Jika pengguna melakukan kesalahan, antarmuka harus menawarkan instruksi yang sederhana, konstruktif, dan spesifik untuk pemulihan.
6. *Permit easy reversal of actions*, Sebisa mungkin, tindakan harus dapat dibalik. Fitur ini mengurangi kecemasan, karena pengguna tahu bahwa kesalahan dapat dibatalkan, dan mendorong eksplorasi opsi yang tidak dikenal.

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



7. *Keep users in control*, Pengguna sangat menginginkan perasaan bahwa mereka mengendalikan antarmuka dan antarmuka merespons tindakan mereka. Mereka tidak menginginkan kejutan atau perubahan dalam perilaku yang sudah dikenal, dan mereka merasa terganggu oleh rangkaian entri data yang membosankan, kesulitan dalam mendapatkan informasi yang diperlukan, dan ketidakmampuan untuk menghasilkan hasil yang diinginkan.
8. *Reduce short-term memory load*, Kapasitas terbatas manusia untuk pemrosesan informasi dalam memori jangka pendek mengharuskan desainer untuk menghindari antarmuka di mana pengguna harus mengingat informasi dari satu tampilan dan kemudian menggunakan informasi itu pada tampilan lain.

G. *Web Browser*

Menurut Didik Setiawan (2020:16), *web browser* memiliki definisi sebagai berikut :

“*Web Browser* adalah sebuah aplikasi yang digunakan untuk menjelajahi situs-situs di dunia maya atau yang biasa disebut dengan *Website*. *Browser* atau *Web Browser* seperti Mozilla, Chrome dan lain sebagainya yang sering kita gunakan untuk menjelajahi situs-situs di dunia maya adalah beberapa contoh dari *Web Browser*”.

Menurut Dendy Kurniawan (2023:8), “*Web Browser*, biasa disebut sebagai browser, adalah alat yang digunakan untuk mengakses, menyimpan, dan mengirimkan informasi di internet”. Selain itu, format yang digunakan untuk informasi tersebut adalah HTML. *Web browser* akan memeriksa kode HTML yang dibuat untuk memastikan bahwa kode tersebut ditampilkan dengan benar.



H. Analisis Perancangan Sistem Informasi

Menurut Scott Tilley (2020:4), “Analisis perancangan sistem informasi adalah proses langkah demi langkah untuk mengembangkan sistem informasi berkualitas tinggi”.

Menurut Joseph S. Valacich, dkk (2015:30), analisis perancangan sistem informasi dapat didefinisikan sebagai berikut :

”Analisis perancangan sistem informasi adalah metode yang digunakan oleh perusahaan seperti IBM, PepsiCo, dan Sony untuk membuat dan mempertahankan sistem informasi yang menjalankan fungsi bisnis dasar, seperti melacak nama dan alamat pelanggan, memproses pesanan, dan membayar karyawan”.

Tujuan utama dari analisis dan desain sistem adalah untuk meningkatkan sistem organisasi. Proses ini sering kali melibatkan pengembangan atau pengadaan perangkat lunak aplikasi serta pelatihan karyawan untuk menggunakannya. Perangkat lunak aplikasi, yang juga disebut sebagai sistem, dirancang untuk mendukung fungsi atau proses organisasi tertentu, seperti manajemen inventaris, penggajian, atau analisis pasar.

Tujuan dari perangkat lunak aplikasi adalah mengubah data menjadi informasi.

I. Penjadwalan

Menurut Michael L. Pinedo (2020:1), penjadwalan dapat didefinisikan sebagai berikut :

“Penjadwalan adalah proses pengambilan keputusan yang digunakan secara teratur di banyak industri manufaktur dan jasa. Proses ini berkaitan dengan alokasi sumber daya untuk tugas-tugas dalam periode waktu tertentu dan tujuannya adalah untuk mengoptimalkan satu atau lebih tujuan”.

Menurut Susmita Bandyopadhyay (2020:369), Penjadwalan dapat didefinisikan sebagai penetapan tanggal mulai dan/atau selesai untuk operasi atau kelompok operasi



guna menunjukkan kapan hal tersebut harus dilakukan agar pesanan manufaktur dapat diselesaikan tepat waktu.

J. Sistem Penjadwalan

Menurut Astri Dwi Juniarahmatunisa (2014:2), “sistem penjadwalan merupakan informasi berupa data dalam proses mengatur rencana kegiatan dengan pembagian waktu pelaksanaan”.

Menurut Rizka dan Mira (2022:724), sistem penjadwalan dapat didefinisikan sebagai berikut:

”Sistem penjadwalan merupakan suatu komponen yang digabungkan untuk mengatur waktu dari suatu kegiatan untuk menentukan pelaksanaan kegiatan dalam suatu organisasi. Sistem penjadwalan bertujuan meminimalkan waktu proses serta penggunaan yang efisien dari fasilitas, tenaga kerja, dan peralatan”.

K. *HyperText Markup Language* (HTML)

Menurut Jennifer Niederst Robbins (2018:10-11), *HyperText Markup Language* (HTML) adalah bahasa yang digunakan untuk membuat dokumen halaman web. HTML bukanlah bahasa pemrograman, melainkan sebuah bahasa *markup*, yang berarti merupakan sistem untuk mengidentifikasi dan mendeskripsikan berbagai komponen dalam sebuah dokumen seperti judul, paragraf, dan daftar.

Menurut Rosa Anamisa dan Ayu Mufarroha (2020:17), *HyperText Markup Language* “(HTML) adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web”. HTML inilah yang menyusun sebuah halaman web menjadi sebagaimana yang kita lihat melalui *browser* (penjelajah internet).



L. Cascading Style Sheets (CSS)

Menurut Rosa Anamisa dan Ayu Mufarroha (2020:39), “CSS adalah bahasa penulisan yang digunakan untuk mendeskripsikan penampilan sebuah dokumen markup”. CSS selalu digunakan bersamaan dengan HTML, meskipun dapat juga digunakan untuk mengatur gaya dokumen *markup* lainnya seperti XML, SVG, dan XUL. HTML dirancang bukan untuk mendesain sebuah halaman web, melainkan hanya untuk menampilkan konten saja.

Berdasarkan penjelasan Rosa Anamisa dan Ayu Mufarroha (2020:43), CSS terdiri dari aturan-aturan gaya yang diinterpretasikan oleh browser dan diterapkan pada elemen-elemen dalam dokumen. Aturan gaya tersebut terdiri dari tiga bagian utama yaitu :

1. *Selector*, adalah tag HTML yang akan diterapkan gaya, seperti <h1> atau <table>
2. *Property*, adalah atribut dari tag HTML, yang dikonversi menjadi properti CSS seperti warna atau border.
3. *Value*, adalah nilai yang diberikan pada properti tersebut, misalnya warna merah atau kode warna seperti #F1F1F1.

M. JavaScript

Menurut Jennifer Niederst Robbins (2018:593), *JavaScript* dapat didefinisikan sebagai berikut:

“*JavaScript* adalah bahasa pemrograman yang menambahkan interaktivitas dan perilaku khusus ke situs web kita. Ini adalah bahasa *client-side scripting*, yang berarti dijalankan di mesin pengguna dan bukan di server, seperti bahasa pemrograman web lainnya seperti PHP dan Ruby.”.

Menurut Rosa Anamisa dan Ayu Mufarroha (2020:67), *JavaScript* merupakan bahasa skrip yang berperan penting di sisi klien. Fungsinya tidak terbatas hanya pada meningkatkan interaksi pengguna dengan halaman web, tetapi juga memungkinkan



pembuatan halaman web menjadi lebih dinamis dan responsif. Dengan *JavaScript*, pengembang dapat menciptakan pengalaman pengguna yang lebih hidup dan interaktif, menambahkan fitur seperti animasi, validasi *form*, pengolahan data secara *real-time*, dan banyak lagi, yang semuanya berkontribusi pada pengalaman pengguna yang lebih baik dan aplikasi web yang lebih kuat secara fungsional.

N. Framework

Menurut Achmad Fikri Sallaby dan Indra Kanedi (2020:593), *Framework* dapat didefinisikan sebagai berikut:

“Framework adalah kumpulan instruksi-instruksi yang dikumpulkan dalam *class* dan *function-function* dengan fungsi masing-masing untuk memudahkan developer dalam memanggilnya tanpa harus menuliskan syntax program yang sama berulang-ulang serta dapat menghemat waktu”.

Menurut Rosa Anamisa dan Ayu Mufarroha (2020:101), *Framework* adalah seperti kerangka besar yang telah disusun sebelumnya, berisi fungsi-fungsi, prosedur-prosedur, dan kelas-kelas yang siap pakai untuk tujuan tertentu. Dengan menggunakan *framework* ini, para pengembang *software* dapat menghemat waktu dan usaha karena mereka tidak perlu membangun semua fitur dari nol. Ini memungkinkan fokus lebih besar pada pengembangan aplikasi yang spesifik dan kompleksitas yang lebih rendah dalam pengembangan software.

O. Laravel

Menurut W. Jason Gilmore (2018:7), Laravel adalah kerangka aplikasi web yang mengambil inspirasi dari fitur-fitur terbaik dari solusi kerangka kerja populer lainnya, seperti *Ruby on Rails* dan ASP.NET MVC. Laravel adalah kerangka kerja pengembangan aplikasi yang cepat. Ini berarti Laravel berfokus pada kurva pembelajaran yang dangkal (mudah) dan meminimalkan langkah-langkah untuk



memulai aplikasi baru dan menerbitkannya. Semua tugas paling umum dalam membangun aplikasi web, mulai dari interaksi basis data hingga autentikasi, antrian, *email*, hingga *caching*, dibuat lebih sederhana oleh komponen-komponen yang disediakan Laravel.

P. Blade

Menurut Matt Stauffer (2019:63), Blade adalah mesin *templating* khusus yang ditawarkan oleh Laravel. Terinspirasi oleh mesin *Razor* milik .NET, *Blade* memiliki sintaks yang ringkas, kurva pembelajaran yang dangkal, model pewarisan yang kuat dan intuitif, serta mudah diperluas. Meskipun PHP berfungsi cukup baik sebagai bahasa *templating* dibandingkan dengan kebanyakan bahasa *backend* lainnya, PHP memiliki kekurangan dan tampilannya yang kurang rapi ketika menggunakan `<?php` secara *inline*. Oleh karena itu, sebagian besar kerangka kerja modern menawarkan bahasa *templating*, dan Blade adalah solusi Laravel untuk kebutuhan ini.

Menurut Jason Gilmore (2018:52), Blade adalah mesin *templating* dari Laravel yang menyediakan sintaks yang disederhanakan untuk menyisipkan logika ke dalam tampilan. Mesin ini memungkinkan penggunaan fitur-fitur seperti pewarisan, pemfilteran output, serta pernyataan kondisional dan perulangan. *Blade* menggunakan ekstensi *.blade.php* untuk mengidentifikasi tampilan yang menggunakan sintaks *Blade*. Mesin ini memfasilitasi penyisipan logika yang diperlukan dalam tampilan, seperti melakukan perulangan pada *array* untuk membuat tabel atau menampilkan isyarat visual jika pengguna menyukai suatu acara. *Blade* menawarkan banyak kemampuan yang menjadikannya mesin *templating* yang kuat dan intuitif.

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



Q. PHP

Menurut Didik Setiawan (2020:54), PHP adalah singkatan dari *Hypertext Preprocessor* yang merupakan bahasa *scripting* yang digunakan pada dokumen HTML.

PHP memiliki sintaks yang hampir sama dengan bahasa C, Java dan Perl. Tujuan dari penggunaan bahasa tersebut adalah untuk memungkinkan perancangan web yang dinamis dan dapat bekerja secara otomatis.

Menurut Kevin Tatroe dan Peter MacIntyre (2020:1), PHP adalah bahasa yang digunakan untuk membuat konten *HyperText Markup Language* (HTML). PHP biasanya dapat digunakan dengan 2 cara yaitu:

1. *Server-side scripting*, PHP awalnya dirancang untuk membuat konten web dinamis yang dapat menghasilkan HTML akan tetapi diperlukan parser PHP dan server web untuk mengirim file dokumen yang sudah dikodekan. Selain itu, PHP juga populer untuk menghasilkan konten dinamis melalui koneksi basis data, dokumen XML, grafik, file PDF, dan lainnya.
2. *Command-line scripting*, PHP dapat menjalankan skrip dari baris perintah, mirip dengan Perl, awk, atau shell Unix. Skrip baris perintah ini dapat digunakan untuk tugas administrasi sistem, seperti pencadangan dan parsing log; bahkan beberapa skrip tipe CRON dapat dilakukan dengan cara ini.

R. Structured Query Language (SQL)

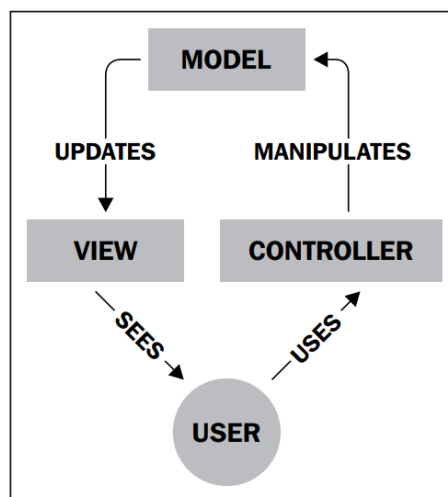
Menurut Ben Forta (2020:1), *Structured Query Language* (SQL) adalah bahasa yang dirancang khusus untuk berkomunikasi dengan basis data. Berbeda dengan bahasa lain, seperti bahasa pemrograman Java, C, atau Python, SQL terdiri dari sedikit kata. Hal ini sengaja dilakukan untuk memberikan cara yang sederhana dan efisien dalam membaca dan menulis data dari basis data.



Menurut Allen G. Taylor dan Richard Blum (2024:1), SQL adalah bahasa standar internasional yang digunakan bersama dengan basis data relasional. Kebetulan, basis data relasional merupakan bentuk dominan penyimpanan data di seluruh dunia.

S. Model View Controller (MVC)

Menurut Arda Kilicdagi dan H. Ibrahim YILMAZ (2014:11), *Model View Controller* (MVC) adalah pola arsitektur dalam rekayasa perangkat lunak yang prinsip dasarnya adalah memisahkan logika aplikasi dari presentasinya. Pola ini membagi aplikasi menjadi tiga bagian yang saling terhubung untuk memisahkan representasi internal informasi dari cara informasi tersebut disajikan atau diterima dari pengguna.



Gambar 2. 2
Ilustrasi Model View Controller
(Sumber: Arda Kilicdagi ,2014:11)

Berdasarkan penjelasan Matt Stauffer (2019:23), *Model View Controller* (MVC) mempunyai 3 konsep utama yaitu

1. *Model*, mewakili tabel database individu atau catatan dari tabel tersebut.
2. *View*, mewakili template yang menampilkan data kepada pengguna akhir. Contohnya seperti halaman login dengan set HTML, CSS, dan JavaScript tertentu.



3. *Controller*, merupakan pengatur lalu lintas seperti mengambil permintaan HTTP dari browser, mengambil data yang tepat dari database dan mekanisme penyimpanan lainnya, memvalidasi input pengguna, dan akhirnya mengirimkan respons kembali ke pengguna.

T. *Marketing*

Menurut Philip Kotler ,dkk (2024:27), Pemasaran adalah kumpulan strategi dan aktivitas yang digunakan perusahaan untuk memperoleh dan melibatkan pelanggan, membangun hubungan yang kuat, dan menciptakan nilai unggul bagi pelanggan sehingga perusahaan dapat menangkap nilai dari pelanggan tersebut. Tujuan utama pemasaran menurut Kotler dibagi menjadi 3 yaitu

1. menarik pelanggan baru dengan menawarkan nilai unggul
2. mempertahankan dan mengembangkan pelanggan saat ini dengan memenuhi janji yang diberikan
3. memastikan profitabilitas jangka panjang dengan memanfaatkan nilai dari pelanggan yang puas dan loyal.

Menurut Budi Rahayu Tanama Putri (2017:1), Pemasaran adalah proses sosial dan manajerial di mana individu dan kelompok mendapatkan apa yang mereka butuhkan dan inginkan dengan menciptakan, menawarkan, dan mempertukarkan produk, jasa, dan nilai dengan orang lain.

U. *Unified Model Language (UML)*

Menurut Grady Booch, dkk (2005:13), *Unified Model Language (UML)* adalah bahasa standar yang digunakan untuk membuat denah perangkat lunak. UML digunakan



untuk memvisualisasikan, menentukan, membangun, dan mendokumentasikan artefak dari sistem yang berfokus pada perangkat lunak.

Menurut Suriya Sundaramoorthy (2022:2), *Unified Model Language* (UML) adalah alat diagram standar untuk desain yang menggunakan bahasa grafis yang berguna untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan artefak sistem perangkat lunak. UML dapat membantu pengembang dan pelanggan memahami perangkat lunak atau sistem yang akan dikembangkan dengan lebih baik.

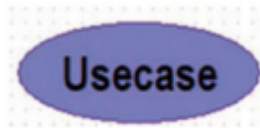
1. Use-case Diagram

Menurut Suriya Sundaramoorthy, (2022:3), *Diagram use-case* berfokus pada identifikasi kebutuhan fungsional dari sistem yang sedang dipertimbangkan. *Use-case diagram* juga memiliki komponen-komponen sebagai berikut ini :

Tabel 2. 1
Komponen-Komponen Use-Case Diagram

Notasi UML	Deskripsi
Batasan Sistem 	Merepresentasikan ruang lingkup sistem dan mencakup seluruh fungsionalitas yang ada dalam sistem tersebut.
Aktor/Actors 	aktor dapat berupa individu, organisasi, atau sistem eksternal yang berinteraksi dengan aplikasi atau sistem Anda.
Use cases	<i>Use case</i> merupakan representasi visual dari fungsionalitas bisnis yang spesifik

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.

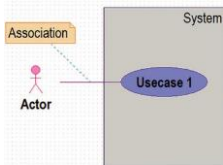
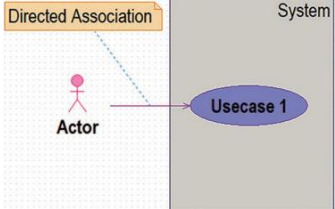
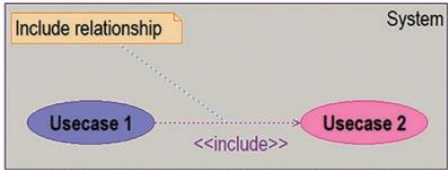
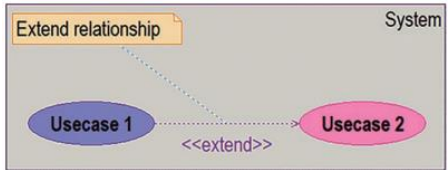


dalam suatu sistem, memastikan bahwa proses bisnis bersifat diskrit.

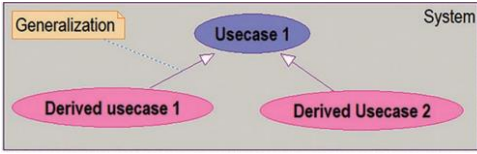
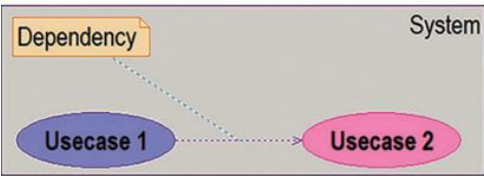
Sumber: Suriya Sundaramoorothy (2022:4)

Selain komponen *use case* juga memiliki *relationship* antara satu komponen dengan komponen lainnya yaitu :

Tabel 2. 2
Hubungan Komponen-Komponen Use Case Diagram

Hubungan	Deskripsi
<i>Association</i> 	hubungan antara aktor dan <i>use case</i> .
<i>Directed Association</i> 	hubungan satu arah di mana aktor bertanggung jawab mempengaruhi <i>use case</i> .
<i>Include</i> 	menggambarkan situasi di mana satu <i>use case</i> mencakup fungsionalitas dari <i>use case</i> lainnya.
<i>Extend</i> 	menunjukkan bahwa <i>use case</i> yang diperluas menambahkan perilaku tambahan pada fungsionalitas yang ada dari <i>use case</i> dasar.



<i>Generalization</i> 	hubungan antara <i>use case</i> induk dan satu atau lebih <i>use case</i> anak.
<i>Dependency</i> 	hubungan di mana keberadaan satu <i>use case</i> bergantung pada keberadaan <i>use case</i> lainnya.

Sumber: Suriya Sundaramoorothy (2022:5)

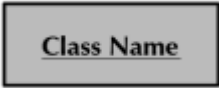
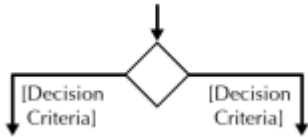
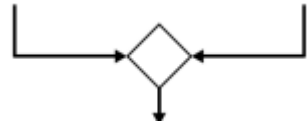
2. Activity Diagram

Menurut Alan Dennis ,dkk (2020b , 2019a :127), Diagram aktivitas umumnya digunakan untuk meningkatkan pemahaman kita terhadap proses bisnis dan model kasus penggunaan yang kita miliki. Diagram aktivitas memiliki notasi-notasi sebagai berikut :

Tabel 2. 3
Komponen-Komponen Activity Diagram

Notasi	Deskripsi
<i>An action</i> 	perilaku sederhana dan tidak dapat diuraikan
<i>An activity</i> 	merepresentasikan kumpulan tindakan



<i>An object node</i> 	merepresentasikan objek yang terhubung dengan serangkaian aliran objek.
<i>A control flow</i> 	menunjukkan urutan dari eksekusi
<i>An object flow</i> 	Menunjukkan aliran dari suatu objek dari satu aktivitas/Tindakan ke aktivitas/tindakan lainnya.
<i>An initial node</i> 	Menggambarkan awal dari serangkaian tindakan atau aktivitas.
<i>A final-activity node</i> 	Digunakan untuk menghentikan semua alur kontrol dan alur objek dalam sebuah aktivitas atau tindakan.
<i>A final-flow node</i> 	Digunakan untuk menghentikan alur kontrol atau alur objek tertentu.
<i>A decision node</i> 	digunakan untuk merepresentasikan kondisi uji untuk memastikan bahwa alur kontrol atau alur objek hanya melanjutkan satu jalur.
<i>A merge node</i> 	Digunakan untuk menggabungkan kembali berbagai jalur keputusan yang telah dibuat menggunakan node keputusan.

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.

<i>A fork node</i> 	Digunakan untuk membagi perilaku menjadi serangkaian alur aktivitas yang paralel atau konkuren.
<i>A join node</i> 	Digunakan untuk menggabungkan kembali serangkaian alur aktivitas yang paralel atau konkuren.
<i>A swimlane</i> 	Digunakan untuk membagi diagram aktivitas menjadi baris dan kolom untuk menugaskan aktivitas individu atau objek yang bertanggung jawab untuk melaksanakan aktivitas tersebut.

Sumber: Alan Dennis (2020b , 2019a:140)

V. *Entity Relationship Diagram (ERD)*

Menurut Alan Dennis dkk., (2019a, 2020b :187), *Entity Relationship Diagram* (ERD) adalah gambaran yang menunjukkan informasi yang dibuat, disimpan, dan digunakan oleh sebuah sistem bisnis. Seorang analis dapat membaca ERD untuk menemukan bagian-bagian informasi individual dalam sistem dan bagaimana mereka diorganisir dan berhubungan satu sama lain. Diagram hubungan entitas tersebut memiliki notasi-notasi sebagai berikut



Tabel 2. 4
Simbol-Simbol Pemodelan Data *Crow's Foot*

Notasi	Deskripsi
ENTITY NAME *Identifier	entitas adalah orang, tempat, atau benda yang memiliki nama tunggal yang ditulis dengan huruf kapital semua, Memiliki identifikasi dan Seharusnya berisi lebih dari satu instansi data.
ENTITY NAME Attribute name Attribute name Attribute name	Atribut adalah properti dari entitas yang digunakan oleh setidaknya satu proses bisnis dan diuraikan menjadi tingkat detail yang paling berguna.
<u>Relationship name</u>	Sebuah hubungan menunjukkan asosiasi antara dua entitas, memiliki entitas induk dan entitas anak, dideskripsikan dengan frasa kerja, memiliki <i>cardinality</i> (1 : 1, 1 : N, atau M : N), memiliki modalitas (<i>null, not null</i>), dan dapat bergantung atau mandiri.

Sumber: Alan Dennis (2019a, 2020b :190)

Berdasarkan penjelasan Carlos Coronel dan Steven Morris (2023: 44), Hubungan menggambarkan asosiasi di antara data. Sebagian besar hubungan menggambarkan asosiasi antara dua entitas Dan terdapat berbagai jenis hubungan dalam data yaitu:



Tabel 2. 5
Simbol-Simbol Model Hubungan data *Crow's Foot*

Notasi	Penjelasan
	Sebuah hubungan satu-ke-banyak (1:m)
	Sebuah hubungan banyak-ke-banyak (m:n)
	Sebuah hubungan satu-ke-satu (1:1)

Sumber: Carlos Coronel dan Steven Morris (2023: 44)

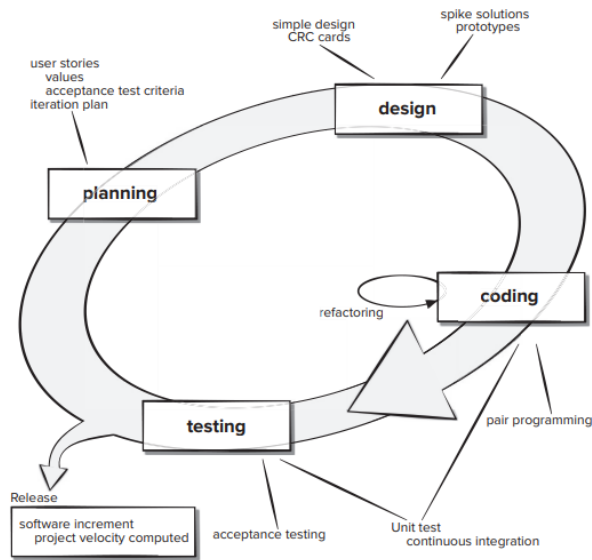
W. *Extreme Programming (XP)*

Menurut Roger S. Pressman dan Bruce R. Maxim (2020:46), *Extreme Programming (XP)* adalah salah satu pendekatan yang paling banyak digunakan dalam pengembangan perangkat lunak *agile*. XP mencakup serangkaian aturan dan praktik yang terjadi dalam konteks empat aktivitas kerangka kerja: perencanaan, desain, pengkodean, dan pengujian. Berikut ini adalah langkah-langkah yang dijelaskan oleh Roger S. Pressman mengenai proses XP yaitu :



C Hak cipta milik IBI KKG (Institut Bisnis dan Informatika Kwik Kian Gie)

Hak Cipta Dilindungi Undang-Undang



Gambar 2. 3
Proses *Extreme Programming*

(Sumber: Roger S. Pressman dan Bruce R. Maxim, 2020:46)

1. **Planning.** Aktivitas perencanaan (juga disebut permainan perencanaan) dimulai dengan aktivitas kebutuhan yang disebut mendengarkan. Dengan mendengarkan maka akan terciptanya serangkaian cerita yang menggambarkan keluaran, fitur, dan fungsionalitas yang diperlukan untuk perangkat lunak yang akan dibangun.
2. **Design,** Desain XP disini mengikuti prinsip KIS (*keep it simple*). Desain fungsionalitas tambahan tidak dianjurkan. XP mendorong penggunaan kartu CRC sebagai mekanisme yang efektif untuk berpikir tentang perangkat lunak dalam konteks berorientasi objek. Kartu CRC (*class-responsibility-collaborator*) mengidentifikasi dan mengorganisir kelas-kelas berorientasi objek yang relevan dengan inkrement perangkat lunak saat ini.
3. **Coding,** Setelah cerita user dikembangkan dan pekerjaan desain awal selesai, tim tidak langsung bergerak ke pengkodean, melainkan mengembangkan serangkaian uji unit yang akan menguji setiap cerita yang akan dimasukkan dalam versi saat ini. Setelah uji unit dibuat, pengembang dapat lebih fokus pada apa yang harus diimplementasikan untuk lulus uji tersebut. Setelah kode selesai,

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



kode tersebut dapat diuji unit secara langsung, sehingga memberikan umpan balik instan kepada para pengembang.



Hak cipta milik IBI KKG (Institut Bisnis dan Informatika Kwik Kian Gie)

Hak Cipta Dilindungi Undang-Undang

4. *Testing*, Test unit yang dibuat harus diimplementasikan menggunakan kerangka kerja yang memungkinkan mereka untuk diotomatisasi sehingga dapat dijalankan dengan mudah dan berulang kali. Hal ini mendorong penerapan strategi pengujian regresi setiap kali kode dimodifikasi.

X. *Black-Box Testing*

Menurut Roger Y. Lee (2019:202), tujuan dari pengujian *black-box* adalah untuk menunjukkan bahwa sebuah komponen beroperasi dengan cara tertentu, yaitu dengan mengubah masukan menjadi keluaran yang dapat diverifikasi. Pengguna hanya perlu mengetahui informasi yang diperlukan untuk bekerja dengan antarmuka serta nilai potensial yang diperlukan. Pemahaman tentang kebutuhan pengguna sangat penting untuk mengetahui keluaran yang diharapkan berdasarkan masukan tertentu. Pengujian *black-box* tidak tertarik pada pemeriksaan sintaks dan struktur logis internal aplikasi, seperti yang dilakukan oleh pengujian *white-box*. Berdasarkan kondisi masukan yang valid atau tidak valid, sistem harus berperilaku sesuai dengan persyaratan fungsional yang telah dirancang.

Y. Penelitian Terdahulu

Pada tahun 2019, Putu Indra Ari Purnama melakukan penelitian yang berjudul “Implementasi Sistem Informasi Rekapitulasi Manajemen Jadwal Kunjungan (Primakara)”. Tujuan dari penelitian ini adalah untuk membangun sistem informasi berbasis web dengan SMS Gateway menggunakan framework Laravel yang dapat mempermudah divisi pemasaran STMIK Primakara dalam melakukan penjadwalan kunjungan ke sekolah-sekolah lain secara terorganisir.

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



Pada tahun 2023, penelitian lainnya dilakukan juga oleh Asril Zaidan Mahruza dan Bambang Agus Herlambang yang berjudul “Implementasi Prototipe Website Penjadwalan Kunjungan Industri Sekolah”. Di dalam penelitian ini sistem dirancang menggunakan Laravel yang sudah disertai dengan dokumentasinya dan menggunakan Mysql sebagai database untuk sistem tersebut. Tujuan dari penelitian ini adalah membangun sistem penjadwalan kunjungan industri di perusahaan tersebut dari berbagai sekolah.

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik dan tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IBIKKG.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.