



B. Sistem

Menurut Scott Tilley (2020:4), sistem didefinisikan sebagai berikut:

“Sistem adalah sekumpulan komponen yang saling berkaitan yang menghasilkan hasil tertentu. Setiap sistem membutuhkan masukan data yang terdiri dari dasar fakta yang merupakan bahan baku bagi sistem”.

Menurut Jeperson Hutahaean (2015:2-7), sistem memiliki definisi sebagai berikut:

“Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu”.

Agar sistem dapat berjalan dengan baik, maka rangkaian prosedur ini harus saling berkaitan dan saling membutuhkan. Sistem memiliki beberapa karakteristik atau sifat yang terdiri dari komponen, batasan sistem (*boundary*), lingkungan luar sistem (*environment*), penghubung sistem (*interface*), masukan sistem (*input*), keluaran (*output*), pengolah sistem dan sasaran sistem. Pada dasarnya sistem dapat diklasifikasikan menjadi 8 klasifikasi dari berbagai sudut pandang:

1. Sistem Abstrak, seperti namanya, sistem ini terbentuk dari pemikiran atau ide-ide yang tidak tampak secara fisik. Misalnya sistem teologi, yaitu sistem yang berupa pemikiran mengenai hubungan manusia dengan Tuhan.
2. Sistem Fisik, merupakan sistem yang ada secara fisik, seperti sistem komputer, sistem perbankan, sistem penjualan, sistem produksi dan lain sebagainya.
3. Sistem Alamiah, sistem yang terjadi atau terbentuk melalui peristiwa alam (*naturally*). Sistem ini tidak dibuat oleh manusia dan tidak bersifat abstrak. Beberapa contoh sistem alamiah seperti sistem perputaran bumi, pergantian siang dan malam atau pergantian musim.
4. Sistem Buatan Manusia, sistem ini melibatkan interaksi manusia dengan mesin yang biasa dikenal dengan sebutan *human machine system*. Sistem informasi merupakan



salah satu contoh dari *human machine system* karena adanya interaksi antara manusia dengan komputer.

5. Sistem Tertentu, sistem yang beroperasi dengan tingkah laku yang sudah dapat diprediksi, sebagai keluaran sistem yang dapat diramalkan.
6. Sistem Tak Tentu, sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilistik.
7. Sistem Tertutup, sistem yang tidak terpengaruh dan tidak berhubungan dengan lingkungan luar. Secara teoritis, sistem tertutup ada namun secara kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada hanya sistem yang relatif tertutup (*relatively closed system*).
8. Sistem Terbuka, sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima input dan output dari lingkungan luar atau sub-sistem lainnya. Sistem ini memerlukan pengendali yang baik karena terbuka dan dipengaruhi oleh lingkungan luar.

C. Informasi

Menurut George (2017:4-6), informasi dapat didefinisikan sebagai berikut:

“Informasi merupakan kumpulan data yang terorganisir dan diproses supaya memiliki nilai tambahan di luar nilai yang dimiliki oleh fakta-fakta secara individu”.

Informasi yang berkualitas berperan penting dalam pengambilan keputusan namun tidak semua data diproses menjadi informasi yang berkualitas. Informasi yang berkualitas dapat dibedakan dengan karakteristik tersebut:

1. Mudah Diakses, informasi harus mudah diakses oleh pengguna-pengguna yang berkepentingan sehingga mereka dapat memperoleh informasi dalam format dan waktu yang tepat untuk memenuhi kebutuhan.



2. Akurat, informasi yang akurat merupakan informasi yang bebas dari kesalahan.

Ⓒ Dalam beberapa kasus, informasi yang tidak akurat dihasilkan dari data yang tidak akurat yang dimasukkan ke dalam proses transformasi. Hal tersebut sering disebut sebagai sampah masuk dan sampah keluar.

3. Lengkap, informasi yang lengkap memuat semua fakta-fakta penting.

4. Ekonomis, informasi harus relatif ekonomis untuk diproduksi. Para pengambil keputusan harus selalu menjaga keseimbangan antara nilai dari informasi dan biaya yang dibutuhkan untuk memproduksi informasi tersebut.

5. Fleksibel, informasi yang fleksibel dapat digunakan untuk berbagai macam kegunaan.

6. Relevan, informasi yang relevan merupakan informasi yang penting untuk para pengambil keputusan.

7. Dapat Diandalkan, informasi yang dipercaya oleh para pengguna. Di banyak kasus, keandalan sebuah informasi bergantung dengan keandalan metode yang digunakan untuk mengumpulkan data. Di kasus lainnya, keandalan bergantung pada sumber informasi.

8. Aman, informasi harus aman dan dijauhkan dari akses pengguna-pengguna yang tidak berkepentingan.

9. Sederhana, informasi harus sederhana dan tidak kompleks. Informasi yang rumit dan detail biasanya tidak dibutuhkan. Faktanya, terlalu banyak informasi menyebabkan kelebihan informasi dimana pengambil keputusan memiliki terlalu banyak informasi dan tidak dapat memilih yang mana yang penting.

10. Tepat Waktu, informasi yang tepat waktu dapat disajikan pada saat dibutuhkan.

Hak Cipta Dilindungi Undang-Undang

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

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.



Gambar diatas menunjukkan 3 pilar utama yang menentukan kualitas informasi. Ketiga pilar tersebut antara lain:

1. Dimensi Waktu, terbagi menjadi 4 sub-pilar, yaitu: *timeliness* (informasi harus tersedia saat dibutuhkan), *currency* (informasi harus selalu terkini saat dibutuhkan), *frequency* (informasi harus disediakan sesering yang diperlukan), dan *time period* (informasi harus tersedia dalam periode waktu yang lalu, waktu sekarang dan waktu yang akan datang).
2. Dimensi Konten, terbagi menjadi 5 sub-pilar, yaitu: *accuracy* (informasi harus terbebas dari *error*), *relevance* (informasi harus terkait dengan informasi yang dibutuhkan sang penerima untuk situasi tertentu), *completeness* (seluruh informasi yang dibutuhkan harus tersedia), *conciseness* (hanya informasi yang diperlukan yang harus disediakan), *scope* (informasi dapat memiliki cakupan yang luas atau sempit, atau berfokus pada internal atau eksternal), *performance* (informasi dapat mengungkap kinerja dengan mengukur aktivitas yang dicapai, kemajuan yang dibuat, atau akumulasi sumber daya).
3. Dimensi Bentuk, terbagi menjadi 5 sub-pilar, yaitu: *clarity* (informasi harus tersedia dalam bentuk yang mudah dimengerti), *detail* (informasi dapat tersedia secara detail ataupun dalam bentuk ringkasan), *order* (informasi dapat diatur dalam urutan yang telah ditentukan), *presentation* (informasi harus dapat dipresentasikan secara naratif, numerik, grafik, atau dalam bentuk lainnya), *media* (informasi dapat tersedia dalam bentuk dokumen fisik, tampilan video, ataupun media lainnya).

D. Sistem Informasi

Menurut Stair dan Renolds (2018:6), sistem informasi dapat didefinisikan sebagai berikut



“Sistem informasi adalah sekumpulan komponen yang saling terkait yang mengumpulkan, memproses, menyimpan dan menyebarkan data serta informasi. Sistem informasi menyediakan mekanisme umpan balik (*feedback*) untuk memantau dan mengendalikan operasinya agar memenuhi tujuan tertentu”.

Menurut Tata Sutabri (2012:46), sistem informasi merupakan:

“Suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan”.

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*building block*). Sebagai suatu sistem, masing-masing blok saling berinteraksi satu dengan yang lain untuk membentuk suatu kesatuan untuk mencapai sasaran. Blok-blok tersebut terdiri dari:

1. Blok Masukan, blok tersebut memiliki input yang mewakili data yang masuk ke dalam sistem informasi. Input yang dimaksud adalah metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.
2. Blok Model, blok ini terdiri dari kombinasi prosedur, logika, dan model matematika yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara tertentu untuk menghasilkan keluaran yang diinginkan.
3. Blok Keluaran, produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.
4. Blok Teknologi, teknologi merupakan “*tool box*” dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari

Hak Cipta Dilindungi Undang-Undang

Hak cipta milik IBI KKG (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.



sistem secara keseluruhan. Teknologi terdiri dari 3 bagian utama, yaitu teknisi (*brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

5. Blok Basis Data, basis data atau *database* merupakan kumpulan data yang saling berkaitan dan berhubungan satu sama lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa agar informasi yang dihasilkan berkualitas.
6. Blok Kendali, banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, ketidak-efisienan, dan sabotase. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila mana terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

E. Analisis dan Perancangan Sistem Informasi

Menurut King (2021:1), analisis dan perancangan sistem informasi dapat didefinisikan sebagai berikut:

“Analisis dan perancangan sistem informasi adalah suatu proses dalam mendefinisikan aspek dari sistem, seperti arsitektur, modul, komponen, dan banyak tampilan antara komponen-komponen tersebut dengan data yang mengalir di dalamnya”.

Sistem ini harus dirancang sedemikian rupa untuk dapat memenuhi tuntutan dan persyaratan khusus bagi perusahaan ataupun organisasi dengan membangun sistem yang berfungsi dengan baik.



Menurut Charles (2005:247), analisis dan perancangan sistem informasi adalah:

“Perancangan dan pengembangan sistem dimulai dengan penurunan dan pengembangan spesifikasi serta persyaratan sistem yang mengikat ruang solusi pengguna dengan tunduk pada biaya, jadwal, analisis, perancangan dan pengembangan sistem”.

Charles (2005:248) juga menekankan bahwa para pengembang harus memahami secara mendalam apa yang ingin dicapai oleh pengguna dan memilih solusi terbaik dari sekumpulan potensi solusi-solusi dengan mempertimbangkan faktor-faktor seperti teknis, teknologi, dukungan, biaya, jadwal, dan risiko untuk mengembangkan sistem yang efektif. Dengan adanya strategi yang baik dan alur kerja pendukung yang menerapkan praktik terbukti memungkinkan sebuah program untuk berkembang dengan efisien dari penyerahan kontrak hingga penyerahan sistem dan penerimaan.

F. Information Sharing System

Menurut Shelley Xin Li dan Tantina Sandino (2017:7), *information sharing system* dapat didefinisikan sebagai berikut:

“Sistem berbagi informasi yang berkualitas dapat mempengaruhi orang-orang yang terlibat di dalamnya. Sistem membantu meningkatkan kreativitas dan keterikatan pengguna saat pengguna lebih sering berinteraksi dengan sistem”.

Sedemikian rupa, dengan adanya sistem pertukaran informasi, jumlah dan kualitas ide-ide yang dihasilkan serta kepuasan partisipan dalam sesi curah pendapat (*brainstorming*) meningkat.

Menurut Maged Abdelsayed dan Ronnie Navon (2014:10), *information sharing system* memiliki definisi sebagai berikut:

“*Information sharing system* atau sistem berbagi informasi merupakan sebuah model yang menggunakan teknologi internet untuk berbagi, mengakses, dan mengelola informasi proyek antara berbagai anggota tim”.



H. Reminder System

Menurut Brewer, dkk. (2017:1), *reminder system* dapat didefinisikan sebagai berikut:

“Sistem pengingat digital digunakan untuk melayani orang-orang sebagai sebuah alat untuk membantu manusia mengingat informasi, atau melakukan suatu tindakan yang jika tidak dilupakan”.

Menurut kamus Webster dari buku Barry Silverman (1997:43), “pengingat merupakan sesuatu yang digunakan untuk membantu mengingatkan seseorang terhadap sesuatu”.

Dengan demikian sistem-sistem pengingat memiliki tujuan utama untuk melawan lupa serta kelalaian yang dimiliki manusia. Jika didefinisikan secara luas, sebuah program pengingat adalah sistem yang memiliki kesadaran pada keberadaannya dan menggunakan pengetahuan yang biasanya diperoleh dari aplikasi lain untuk membantu penggunanya mengingat. Hal tersebut dapat berupa sebuah kejadian, aksi, prosedur penting, metode, dan pengetahuan-pengetahuan yang dapat diaplikasikan ke suatu tugas. Pengguna memiliki kebebasan untuk mengubah, meneruskan, membatalkan, menugaskan kembali, atau mengabaikan pengingat tersebut.

Sistem pengingat digital membantu manusia dalam 2 area memorinya, yakni:

1. *Prospective Memory*, layanan-layanan dari asisten digital seperti Siri, Cortana dan Google Assistant menjadi alat yang efektif dalam membantu ingatan prospektif atau ingatan untuk masa depan yang meliputi tempat dan waktu. Aplikasi pendaftaran digital dan kalender digital membantu para pengguna untuk mengingat peristiwa-peristiwa penting dalam hidupnya.
2. *Retrospective Memory*, layanan-layanan yang membantu dalam mengingat kejadian-kejadian yang sudah terjadi. Sistem pencatatan digital menjadi salah satu layanan yang memperbolehkan penggunanya untuk mengumpulkan konten-konten yang mereka ingin ingat. Teknologi foto digital juga berperan dalam membantu



restrospective memory dimana orang-orang dapat mengingat kembali kejadian terkait terekamnya sebuah gambar.

I. Kerangka Kerja (*Framework*)

Menurut STMIK STIKOM Indonesia (2018:17), "*Framework* atau kerangka kerja aplikasi menyajikan kelas-kelas yang dapat digunakan untuk mengembangkan aplikasi". Selain itu, kerangka kerja juga menyediakan abstraksi generik yang digunakan untuk mengakses perangkat, mengatur antarmuka tampilan, serta sumber daya aplikasi. Bagian terpenting dalam kerangka kerja aplikasi Android adalah sebagai berikut:

1. *Activity Manager*, digunakan untuk mengontrol siklus hidup sebuah aplikasi dan menjaga *backstack* (aktivitas-aktivitas Android yang tersimpan dalam suatu tumpukan atau *stack*) untuk navigasi penggunaan.
2. *Content Providers*, berfungsi untuk merangkum data sehingga memungkinkan untuk digunakan oleh aplikasi lainnya seperti daftar nama.
3. *Resource Manager*, mengatur sumber daya dalam sebuah program serta menyediakan akses sumber daya diluar kode program seperti karakter, grafik, dan tata letak berkas.
4. *Location Manager*, berfungsi untuk memberikan informasi rinci mengenai lokasi dimana perangkat Android berada.
5. *Notification Manager*, menjangkau berbagai macam pengingatan seperti pesan masuk, janji, dan kejadian tertentu yang ditampilkan pada *status bar*.

J. *Application Programming Interface (API)*

Menurut Brenda Jin, dkk. (2018:1), API memiliki definisi sebagai berikut:



“*Application Programming Interface* atau sering disingkat menjadi API merupakan antarmuka dari program perangkat lunak yang dipresentasikan kepada program lain, manusia, dan jika berbasis web, ke dunia via internet. API mengizinkan pertukaran informasi dalam *platform* bisnis yang ada di internet dan menjadi sebuah identitas yang terintegrasi lintas perangkat lunak”.

Menurut Daniel Jacobson, dkk. (2012:7), API memiliki sifat sebagai berikut:

“Sifat API dapat diklasifikasikan menjadi 2 tipe yaitu API publik dan privat. Publik berarti API dapat diakses oleh hampir semua orang tanpa adanya persetujuan dalam kontrak (diluar syarat penggunaan) dengan penyedia API. Sedangkan API yang sifatnya privat digunakan dalam berbagai cara, yaitu dengan mendukung upaya internal API atau penggunaan API oleh mitra dan biasanya memiliki kontrak secara hukum. Biasanya perusahaan memula dengan API privat namun ujungnya akan membuka sebagian untuk akses publik dengan batasan-batasan tertentu”.

K. Basis Data (*Database*)

Menurut Elvis Foster (2022:3), basis data memiliki definisi sebagai berikut:

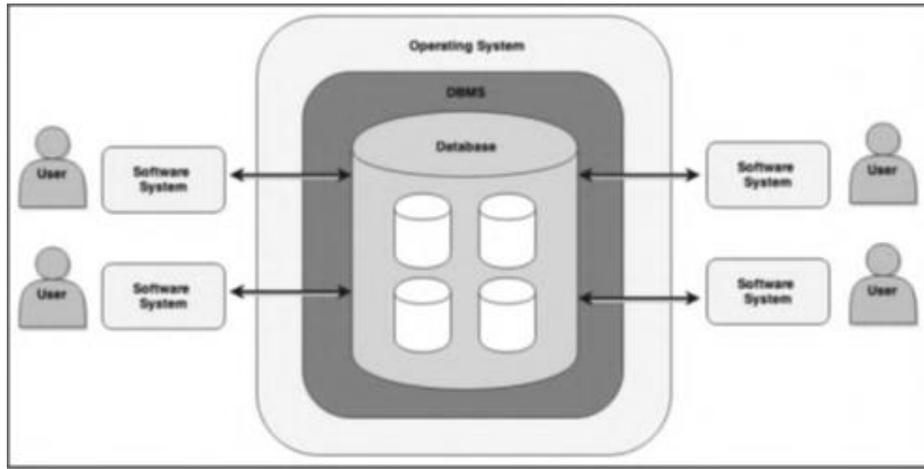
“*Database system* (DBS) merupakan sistem pencatatan yang terkomputerisasi dengan tujuan untuk mengelola informasi dan membuatnya tersedia ketika informasi tersebut dibutuhkan”.

Menurut Allen (2013:7), basis data dapat didefinisikan sebagai berikut:

“Basis data atau *database* merupakan koleksi arsip-arsip terintegrasi yang dapat menggambarkan diri sendiri. Dalam teknologi komputer basis data biasanya diasosiasikan dengan bahasa pemrograman seperti SQL. Sebuah *record* merupakan representasi dari sebuah objek fisik dan konseptual. Basis data terdiri dari data dan *metadata* yang merupakan data yang menjelaskan struktur data dalam sebuah basis data”.

Basis data biasanya menyimpan data dalam sebuah sistem komputer dengan bantuan *Database Management System* (DBMS) yang merupakan serangkaian program yang menyediakan layanan untuk mengelola sebuah basis data. Komponen-komponen yang mendukung sebuah basis data meliputi perangkat keras dan sistem operasi, *Database Management System*, basis data, perangkat lunak sistem atau aplikasi yang terkait, dan pengguna (pengguna akhir dan teknis).

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 menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



Gambar 2. 2
Ilustrasi Sederhana DBS
(Sumber: Elvis Foster, 2022:4)

Dari penjelasan Elvis Foster (2022:6), sebuah basis data memiliki tujuan utama sebagai berikut:

1. Keamanan dan proteksi, mencegah akses dari para pengguna yang tidak berkepentingan dan proteksi dari gangguan antar proses.
2. Reliabilitas, memastikan adanya performa yang stabil dan dapat diprediksi.
3. Memfasilitasi banyak pengguna.
4. Fleksibilitas, kemampuan untuk memperoleh data dan efek dari aktivitas tersebut dengan menggunakan metode yang berbeda-beda.
5. Kemudahan dalam mengakses dan mengubah data.
6. Akurasi dan konsistensi.
7. Kejelasan, memiliki standarisasi data untuk mencegah ambiguitas.
8. Kemampuan untuk melayani permintaan yang tidak diantisipasi.
9. Proteksi Dari Investasi, biasanya dapat dipenuhi dengan membuat cadangan atau *backup* lalu melakukan prosedur pemulihan.
10. Meminimalisir Proliferasi Data, kebutuhan aplikasi baru mungkin memerlukan data yang sudah ada dibandingkan membuat semuanya dari awal.

Hak Cipta Dilindungi Undang-Undang

© Hak cipta milik IBI KKG (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.



menjadi serba otomatisasi telah mencapai titik puncaknya. Generasi web 4.0 ini berbasis kecerdasan buatan dan kemampuan *cloud computing*. Nova Spivack, pendiri Radar Networks, mendefinisikan web 4.0 sebagai “kemampuan untuk bekerja dengan alat hanya secara daring”.

Menurut Ben Shneiderman, dkk (2017:96), “Terdapat delapan prinsip aturan emas yang berlaku di sebagian besar sistem interaktif, antara lain:

1. *Strive for Consistency*, urutan tindakan yang konsisten harus diperlukan dalam situasi yang serupa; terminologi yang sama harus digunakan dalam *prompt*, menu, layar bantuan, warna, tata letak, ukuran huruf, jenis *font*, dan sebagainya harus digunakan di semua bagian halaman web. Hal ini berguna agar *user* tetap dapat mengenali halaman yang dilihat masih dalam lingkup atau masih memiliki hubungan dengan aplikasi web yang digunakan.
2. *Seek Universal Usability*, yaitu dengan mengenali kebutuhan *user* yang beragam dan memberikan fasilitas transformasi konten. Poin ini mengarah kepada penerapan pada variasi tingkat pemahaman *user* terhadap aplikasi.
3. *Offer Informative Feedback*, setiap tindakan yang dilakukan *user* harus memiliki antarmuka umpan balik (*feedback interface*). Untuk tindakan yang sering dilakukan dan bersifat sederhana, maka respon akan ditampilkan secara sederhana pula. Dengan demikian, *user* mengerti bahwa aksinya sudah direspon oleh aplikasi.
4. *Design Dialogue to Yield Closure*, berupa rangkaian aksi yang harus diatur ke dalam 3 blok utama, yaitu blok awal, blok tengah, dan blok akhir. Sebagai contoh, situs web *e-commerce* mengalihkan pengguna dari pemilihan produk ke halaman pembayaran, kemudian diakhiri dengan halaman konfirmasi yang mengindikasikan transaksi telah selesai.



5. *Prevent Errors*, sebisa mungkin, rancang antarmuka agar pengguna tidak dapat membuat kesalahan fatal. Jika *user* membuat *error*, antarmuka harus menawarkan instruksi yang sederhana, konstruktif dan spesifik untuk melakukan pemulihan. Tindakan yang salah harus membuat antarmuka tidak berubah atau antarmuka harus memberikan instruksi tentang pemulihan keadaan.
6. *Permit Easy Reversal of Actions*, sebisa mungkin, aksi harus dapat dibatalkan. Fitur ini mengurangi kecemasan, karena *user* tahu bahwa kesalahan dapat diurungkan. Dengan demikian, *user* merasa nyaman saat mencoba melakukan eksplorasi pada aplikasi.
7. *Keep Users in Control*, *user* yang berpengalaman dalam menggunakan aplikasi web biasanya ingin bertanggung jawab atas antarmuka yang tersedia. Mereka tidak ingin ada perubahan pada perilaku sistem yang biasa. Dengan kata lain, *user* ingin memiliki tampilan antarmuka yang bisa diatur oleh nya sendiri sesuai preferensi mereka.
8. *Reduce Short-term Memory Load*, kapasitas manusia yang terbatas dalam memproses informasi jangka pendek mengharuskan perancang menghindari antarmuka di mana pengguna harus mengingat informasi dari satu tampilan kemudian menggunakan informasi itu di tampilan lain. Data yang harus dimasukkan seharusnya sudah disediakan oleh sistem”.

N. Web Browser

Menurut Didik Setiawan (2017:16), *web browser* dapat didefinisikan sebagai berikut:

“*Web browser* ialah sebuah aplikasi yang digunakan untuk menjelajahi situs-situs di dunia maya yang biasa disebut dengan *website*. Saat ini terdapat banyak sekali jenis-jenis browser yang dapat digunakan, seperti Mozilla, Chrome, Microsoft Edge, Opera dan Safari”.



O. Firebase

Menurut Ashok Kumar (2018:1), Firebase dapat didefinisikan sebagai berikut:

“Firebase merupakan layanan balik layar yang memiliki berbagai banyak kemampuan dan teknologi yang pada akhirnya di akuisisi Google pada tahun 2014. Firebase telah berubah menjadi sebuah alat yang mendukung hampir semua kasus dunia nyata dalam pengembangan perangkat lunak. Firebase membagi serangkaian layanan yang ada di dalamnya menjadi 3 kategori: untuk mengembangkan aplikasi, meningkatkan kualitas aplikasi, dan membantu pertumbuhan dalam aspek bisnis”.

Dalam buku Laurence Moroney (2017:1), Firebase didefinisikan sebagai berikut:

“Firebase bukan pengganti Application Programming Interface (API) namun merupakan kemajuan yang menyediakan layanan-layanan bagi para pengembang seperti basis data, autentikasi, dan pengiriman pesan. Berbagai layanan tersebut disediakan agar para pengembang dapat fokus pada pengembangan aplikasi”.

Beberapa teknologi yang ditawarkan oleh Firebase memiliki sebuah koneksi yang disimpulkan oleh Google Analytics. Sebagian besar layanan yang ditawarkan oleh Firebase tidak dikenakan biaya. Firebase dikembangkan dari 3 pilar utama yaitu, pengembangan, pertumbuhan, dan penghasilan.



Gambar 2.3
Pilar-Pilar dalam Firebase
(Sumber: Laurence Moroney, 2017:2)

Hak Cipta Dilindungi Undang-Undang
Instititut Bisnis dan Informatika Kwik Kian Gie

Instititut 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.

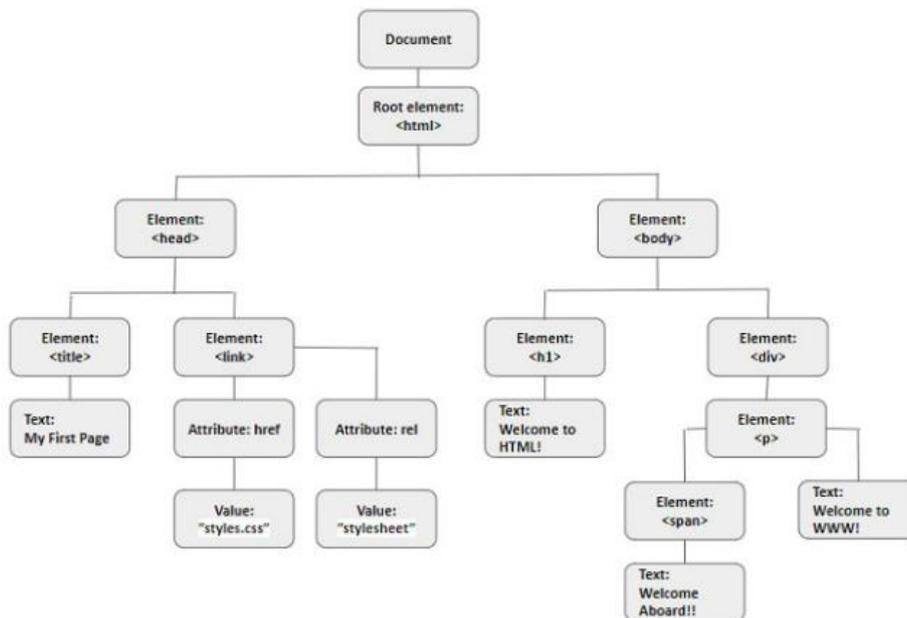


P. HyperText Markup Language (HTML)

Menurut Alok Ranjan, dkk (2020:43), HTML memiliki definisi sebagai berikut:

“HTML adalah bahasa web yang digunakan web browser untuk menulis teks, audio, gambar, video, grafik dan materi lainnya ke dalam halaman web. HTML sebenarnya bukan bahasa pemrograman, melainkan bahasa *markup*. Bahasa *markup* dirancang untuk definisi, presentasi, dan pemrosesan teks. Kode dalam bahasa *markup* pada umumnya dirancang sedemikian rupa sehingga dapat dibedakan secara sintaksis”.

Dokumen HTML terdiri dari struktur seperti pohon. Level pertama berupa tag HTML, yaitu elemen *root* yang terdiri dari dua elemen utama: *head* dan *body*. Tag *head* adalah bagian opsional yang mendefinisikan detail suatu halaman seperti judul, informasi metadata, tautan *styling*, *script* dan *base*. Tag *body* adalah bagian yang berisi konten sebenarnya yang ditampilkan pada sebuah halaman. *Body* dapat berisi sejumlah elemen pendukung lain yang semuanya akan ditampilkan di layar.



Gambar 2. 4
Pohon elemen DOM pada HTML
(Sumber: Alok Ranjan, 2020:48)

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.



Menurut Didik Setiawan (2017:16), HTML dapat didefinisikan sebagai berikut:

- © Hak cipta milik IBIKKG (Institut Bisnis dan Informatika Kwik Kian Gie)
- Hak Cipta Dilindungi Undang-Undang
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.
- “HTML atau *Hyper Text Markup Language* merupakan sebuah bahasa pemrograman terstruktur yang dikembangkan untuk membuat halaman website yang dapat diakses atau ditampilkan menggunakan Web Browser”.

Q. JavaScript

Menurut Alok Ranjan, dkk (2020:118), JavaScript dapat didefinisikan sebagai berikut:

“JavaScript adalah otak di balik semua keindahan konten web. JavaScript menentukan bagaimana interaksi pengguna akan ditangani, apa tindakan yang akan terjadi, dan apa yang akan menjadi status aplikasi selanjutnya”.

Menurut Blum (2018:187), JavaScript memiliki definisi sebagai berikut:

“JavaScript adalah kode program yang disematkan ke dalam kode HTML. *Server* web mengirimkan kode program JavaScript ke web browser, lalu kemudian akan mendeteksi dan menjalankan kode JavaScript”.

JavaScript (JS) adalah bahasa pemrograman komputer ringan, lintas platform, berorientasi objek, yang dikembangkan untuk menambah logika dan perilaku sebuah web.

Menggunakan JavaScript, situs web menjadi lebih dinamis dan interaktif. Kekuatan JavaScript juga terletak pada fakta bahwa JavaScript dapat digunakan untuk pemrograman sisi klien dan sisi *server* karena dapat dijalankan di browser atau *server*, atau di perangkat apapun yang memiliki program spesial yang disebut *JavaScript engine*.

Sebagai sebuah bahasa pemrograman, JavaScript telah distandarisasi oleh ECMA International. Perusahaan ini berdedikasi untuk menciptakan standar untuk komunikasi dan pemrosesan informasi. Salah satu standar yang paling terkenal adalah ECMAScript, yaitu standar bahasa JavaScript.



R. Cascading Style Sheets (CSS)

Menurut Alok Ranjan, dkk (2020:43), CSS dapat didefinisikan sebagai berikut:

“CSS mengambil kendali penuh untuk menata web dan memperkuat konten dengan desain visual, yang kekuatannya tidak terbatas. Ini meningkatkan intuisi web sekaligus meningkatkan tampilan halamannya. CSS adalah keterampilan mutlak yang harus dimiliki dalam pengembangan web”.

Menurut Ummy Gusti Salamah (2021:71), CSS memiliki definisi sebagai berikut:

“CSS merupakan singkatan dari Cascading Style Sheets. Sesuai dengan namanya, CSS memiliki sifat *“style sheet language”* yang berarti bahasa pemrograman yang digunakan untuk *web design*. Dengan menggunakan CSS, maka kita dapat menghasilkan tampilan web yang indah dan menarik. Bahasa pemrograman ini lebih mudah dan ringkas daripada menggunakan atribut di tiap tag HTML. CSS dapat digunakan untuk menciptakan web dengan tampilan yang dinamis dan fleksibel”.

Dengan *style sheets*, Anda mampu mengontrol tata letak beberapa halaman web sekaligus dan membuat situs web menjadi konsisten. Selain itu, Anda juga dapat menerapkan *styling* yang berbeda ke konten yang sama untuk mengubah tampilan di perangkat yang berbeda secara mudah. Cascading Style Sheet (CSS) sudah didukung oleh hampir semua web browser karena CSS telah distandarkan oleh World Wide Web Consortium (W3C).

S. SQL

Menurut Alan Beaulieu (2020:17), “SQL adalah bahasa untuk menghasilkan, memanipulasi, dan menerima data dari *database* relasional”. Salah satu alasan mengapa *database* relasional begitu populer adalah karena *database* relasional yang dirancang dengan baik dapat menangani data dalam jumlah yang besar. Sementara SQL dimulai sebagai bahasa pemrograman yang digunakan untuk memanipulasi data dalam *database* relasional, SQL



kini telah berkembang menjadi bahasa untuk memanipulasi data di berbagai teknologi *database*.

Bahasa SQL dipecah menjadi beberapa bagian yang berbeda, yaitu:

1. *SQL Schema Statements*, digunakan untuk mendefinisikan struktur data yang disimpan dalam database (seperti *table*, *indexes*, atau *constraints*) secara kolektif.
2. *SQL Data Statements*, digunakan untuk memanipulasi struktur data yang telah didefinisikan sebelumnya menggunakan *SQL schema statements*.
3. *SQL Transaction Statements*, yang digunakan untuk memulai, mengakhiri, dan memutar kembali transaksi.

T. Unified Modeling Language (UML)

Menurut Gerard O'Regan (2017:199), UML dapat didefinisikan sebagai berikut:

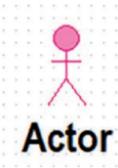
“*Unified Modeling Language (UML)* adalah bahasa pemodelan grafis ekspresif untuk memvisualisasikan, menentukan, membangun dan mendokumentasikan sistem perangkat lunak. UML adalah cara untuk memodelkan sistem perangkat lunak sebelum mengimplementasikannya ke dalam bahasa pemrograman. Hal ini memberikan beberapa tampilan arsitektur perangkat lunak dan memiliki sintaks serta semantik yang jelas”.

1. *Use-case Diagram*

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



Tabel 2. 1
Notasi Pada Use-Case Diagram

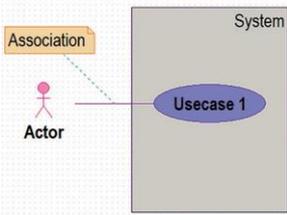
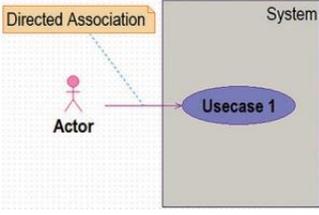
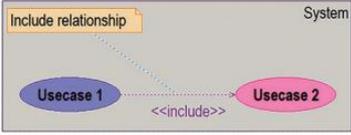
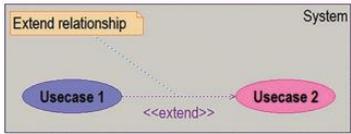
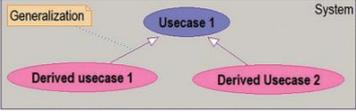
Simbol	Deskripsi
<p>Batasan sistem / <i>system boundary</i></p> 	<p>Merepresentasikan ruang lingkup dan merangkum kumpulan fungsi suatu sistem.</p>
<p>Aktor / <i>actors</i></p> 	<p>Pengguna yang dapat berupa orang, organisasi, ataupun sistem eksternal yang berinteraksi dengan sistem.</p>
<p><i>Usecases</i></p> 	<p>Representasi visual dari fungsionalitas suatu bisnis berbeda dalam suatu sistem.</p>

Sumber: Dr. Suriya Sundaramoorothy (2022:3)

Selain memiliki komponen, *use-case diagram* juga memiliki *relationship* antar satu komponen dengan komponen lain, antara lain:

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.

Tabel 2. 2
Relationship Pada Class Diagram

Relationship	Deskripsi
<p>Asosiasi / <i>association</i></p> 	<p>Hubungan antara aktor dan <i>usecase</i>.</p>
<p>Asosiasi langsung / <i>directed association</i></p> 	<p>Hubungan satu arah dimana aktor memiliki tanggung jawab untuk mempengaruhi <i>usecase</i>.</p>
<p><i>Include</i></p> 	<p>Situasi dimana satu <i>usecase</i> menyertakan fungsi dari <i>usecase</i> lainnya.</p>
<p><i>Extend</i></p> 	<p>Menyiratkan hubungan bermakna antar dua <i>usecase</i> dimana <i>extended usecase</i> menambahkan perilaku baru terhadap fungsi yang telah ada pada <i>usecase</i> pokok.</p>
<p><i>Generalization</i></p> 	<p>Hubungan antara induk <i>usecase</i> dan satu atau lebih <i>child usecases</i>.</p>

Hak Cipta milik IBIKKG (Institut Bisnis dan Informatika Kwik Kian Gie)

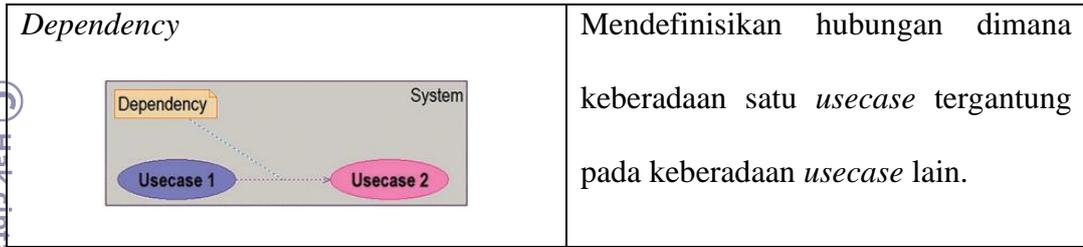
Institut Bisnis dan Informatika Kwik Kian Gie

Hak Cipta Dilindungi Undang-Undang

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 mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IBIKKG.



Sumber: Dr. Suriya Sundaramoorthy (2022:3)

Menurut Rosa dan Shalahuddin (2016:155), diagram *use case* dapat didefinisikan sebagai berikut:

“*Use case* atau diagram *use case* merupakan pemodelan untuk perilaku (*behaviour*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat”.

2. Activity Diagram

Menurut Rosa dan Shalahuddin (2016:161), *activity diagram* dapat didefinisikan sebagai berikut:

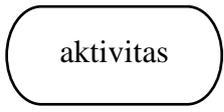
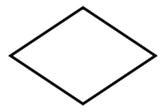
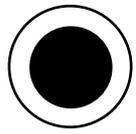
“Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat digunakan oleh sistem”.

Diagram aktivitas memiliki notasi-notasi sebagai berikut:

Tabel 2. 3
Notasi Pada Activity Diagram

Simbol	Deskripsi
	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.



<p>Aktivitas</p> 	<p>Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.</p>
<p>Percabangan /<i>decision</i></p> 	<p>Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.</p>
<p>Penggabungan /<i>join</i></p> 	<p>Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.</p>
<p>Status akhir</p> 	<p>Status akhir yang dilakukan oleh sistem, sebuah diagram aktivitas memiliki sebuah status akhir.</p>

Sumber: Rosa dan Shalahuddin (2016:162)

Hak Cipta Dilindungi Undang-Undang

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

Institut Bisnis dan Informatika Kwik Kian Gie

U. Entity Relationship Diagram (ERD)

Menurut Bagui dan Earp (2023:7), ERD dapat didefinisikan sebagai berikut:

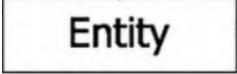
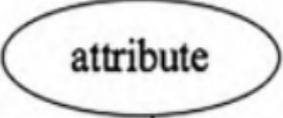
“Sistem basis data sering dimodelkan menggunakan diagram hubungan entitas sebagai kerangka kerja terperinci dimana basis data sebenarnya dibuat. Diagram ER adalah alat pemodelan data semantik yang digunakan untuk mencapai tujuan mendeskripsikan atau menggambarkan data secara abstrak”.

Data yang dideskripsikan secara abstrak disebut model konseptual. Diagram ER juga dapat digunakan untuk mendokumentasikan basis data yang ada dengan merekayasa balik data tersebut. Diagram hubungan entitas memiliki notasi-notasi sebagai berikut:

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.



Tabel 2. 4
Notasi Pada Entity Relationship Diagram

Simbol	Deskripsi
Entitas / <i>entity</i> 	Dapat merepresentasikan objek dunia nyata apapun yang datanya dapat disimpan ke dalam basis data.
Atribut / <i>attribut</i> 	Sifat / <i>properties</i> yang dimiliki oleh tiap-tiap entitas.
Hubungan / <i>relationship</i> 	Mendefinisikan bagaimana dua atau lebih entitas berinteraksi satu sama lain.

Sumber: Diolah Penulis

Pada dasarnya, *relationship* dalam ERD dapat dibagi menjadi beberapa kategori, antara lain:

1. *One to One* (1:1), menandakan hubungan satu arah karena satu entitas dikaitkan dengan satu entitas lain dan sebaliknya.
2. *One to Many* (1:M), memiliki arti satu entitas dapat memiliki relasi dengan beberapa entitas, begitu pula sebaliknya.
3. *Many-to-Many* (M:N), hubungan ini menandakan banyak kejadian dari satu entitas yang dikaitkan dengan banyak kejadian dari entitas lain.

V. Single Entry Method

Menurut Edward dan Walker (2009:105), *single entry method* dapat didefinisikan sebagai berikut:

1. Ditarang 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.

© Hak Cipta milik IBIKKG (Institut Bisnis dan Informatika Kwik Kian Gie)



“Entri tunggal memiliki keunggulan dibandingkan entri ganda karena entri ganda lebih kompleks, dan dengan demikian memungkinkan kerahasiaan yang lebih besar dan lebih banyak ruang untuk penipuan. *Single Entry Method* adalah sebuah metode alternatif untuk melakukan pencatatan akuntansi sederhana untuk *Small and Medium Enterprise*, terutama untuk usaha berskala mikro atau bisnis kecil”.

Mengutip Kohler dari buku Siddiqui (2007:642), “*Single Entry System* adalah sebuah sistem pembukuan dimana sebagai aturan hanya catatan kas dan rekening pribadi yang dipertahankan”.

W. Extreme Programming (XP)

Menurut Roger Pressman (2020:46), *Extreme Programming* dapat didefinisikan sebagai berikut:

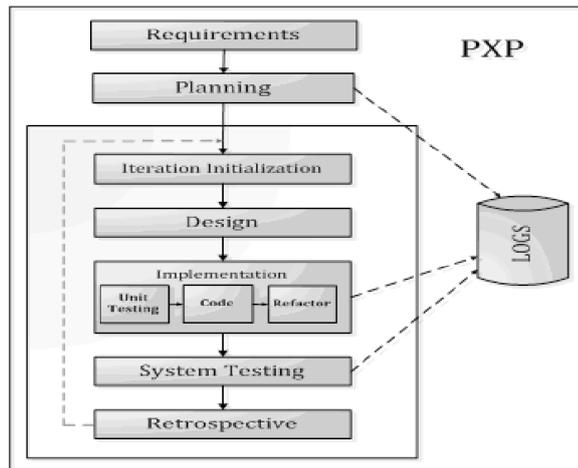
“*Extreme Programming* (XP) merupakan salah satu pendekatan yang sering digunakan dalam pengembangan perangkat lunak. *Extreme Programming* mencakup seperangkat aturan dan praktik yang terjadi dalam konteks empat aktivitas kerangka kerja yaitu perencanaan, desain, pemrograman, dan pengujian”.

Menurut Dzhurov (2009:253), *Personal Extreme Programming* memiliki definisi sebagai berikut:

“*Personal Extreme Programming* (PXP) merupakan metodologi yang bercabang dari *Extreme Programming* yang dirancang untuk meningkatkan performa dan kualitas secara otonom dengan mengotomatisasi aktivitas pengembang harian dalam melakukan retrospeksi secara teratur”.

Berikut adalah fase-fase yang ada dalam PXP menurut Dzhurov:

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.



Gambar 2.5
Fase-Fase Proses dalam PXP
(Sumber: Dzhurov, 2009)

Berikut adalah langkah-langkah yang perlu diketahui dalam menerapkan metode

Personal Extreme Programming (PXP):

Tabel 2.5
Langkah dan Penjelasan Dalam Penerapan PXP

No.	Langkah	Penjelasan
1.	<i>Requirements</i>	Pada tahap kebutuhan, dokumen dengan kebutuhan fungsional dan nonfungsional untuk sistem dibuat. Tahap ini bersifat opsional.
2.	<i>Planning</i>	Selama fase ini, pengembang membuat daftar tugas berdasarkan dokumen persyaratan. Setiap tugas dapat dibagi menjadi tugas yang lebih kecil dengan estimasi waktu berdasarkan data masa lalu atau asumsi terbaik. Keputusan desain seperti bahasa pemrograman, kerangka pengembangan, dan model aplikasi juga dibuat selama fase ini.

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

Hak Cipta Dilindungi Undang-Undang

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.	<i>Iteration Initialization</i>	Setiap iterasi dimulai dengan pemilihan tugas yang akan menjadi fokus iterasi. Pengembang berhak untuk menentukan fokus tersebut.
4.	<i>Design</i>	Pengembang memodelkan sistem yang akan diimplementasikan dalam iterasi yang sedang berlangsung. Pengembang bertujuan untuk mendesain sistem memenuhi kebutuhan pengguna.
5.	<i>Implementation</i>	Pengembang mengimplementasikan semua objek yang didefinisikan dalam fase desain sebelumnya dan menguji mereka. Fase ini terdiri dari tiga sub-fase: pengujian unit, pembuatan kode, dan perbaikan kode, yang dieksekusi sesuai dengan urutannya.
6.	<i>System Testing</i>	Memastikan bahwa kode dapat dikompilasi tanpa kesalahan dan semua pengujian unit harus berhasil untuk keluar dari fase ini.
7.	<i>Retrospective</i>	Pengembang mengevaluasi dan mengembangkan aplikasi secara terus menerus serta melakukan pemeliharaan sistem. Pengembang menganalisis semua proposal perbaikan proses dan, jika perlu, ia dapat menyesuaikan praktik yang menyertainya. Perubahan yang mengatasi masalah dalam proses dan praktik harus terjadi pada tahap awal untuk mencegah kegagalan proyek.

Sumber: Dzhurov (2009)



X. Penelitian Terdahulu

Pada tahun 2008, Jayasinghe, Leelamanie, Yoshida dan Machida melakukan penelitian yang berjudul “*Development of a Fieldwork Reminder System to Help Field Management*”.

Penelitian tersebut dilakukan melihat situasi perkembangan internet dan *Information Communication Technology (ICT)* telah membuka ruang untuk meningkatkan produksi pangan melalui bantuan sistem dengan mengembangkan aplikasi berbasis internet kepada petani-petani kecil untuk mengingatkan pekerjaan lapangan mereka pada waktunya. Aplikasi web yang dikembangkan dalam penelitian tersebut juga berpotensi memberikan pesan pengingat kepada penggunanya di ladang manapun. Dengan begitu, petani dapat melakukan pekerjaannya tanpa menunda jadwal mereka.

Pada tahun 2016, Jaspaljeet Singh Dhillon et al, melaksanakan penelitian dengan judul “*Designing a Web-based telehealth System for Elderly People: An Interview Study in New Zealand*”. Penelitian tersebut secara khusus berfokus pada perancangan aplikasi berbasis web di sektor kesehatan untuk para lansia. Banyaknya lansia yang setuju bahwa dukungan keluarga adalah faktor yang penting membuat peneliti berusaha untuk menyertakan anggota keluarga pengguna, lingkaran sosial dan kelompok pendukung untuk dapat berinteraksi dengan sistem, misalnya dengan memberi sebuah *email* peringatan mengenai aktivitas yang dilakukan oleh lansia yang dapat dikirim ke anggota keluarga, sehingga anggota keluarga dapat melakukan pemantauan langsung terhadap lansia di dalam keluarga tersebut.