

BAB II

LANDASAN TEORI

2.1. Tinjauan Pustaka

Penelitian yang berkaitan dengan website alumni ini diteliti oleh Mohammad Arifin Nurul Qhomar, Dwi Retnoningsih, Agus Purwo Handoko pada tahun 2018 yang berjudul “Membangun Website Komunitas Ikatan Alumni Universitas Sahid Surakarta”. Penelitian ini menjelaskan bahwa dalam menyimpan data kelulusan alumni masih menggunakan bantuan aplikasi *Microsoft Office Excel*. Metode pengembangan sistem yang digunakan adalah *DFD*. Hasil dari penelitian berupa website sebagai media informasi alumni.

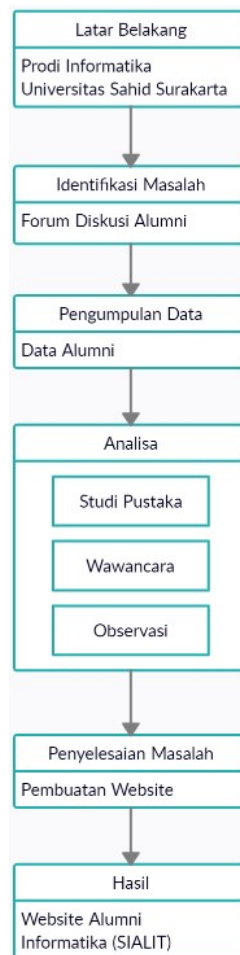
Penelitian lainnya yang berkaitan dengan website alumni ini diteliti oleh Bakhrul Huda Solihin, Fachriyadi Wajaya, Muhammad Ropianto pada tahun 2020 yang berjudul “Perancangan Website Alumni Uis Berbasis Semantic”. Penelitian ini menjelaskan bahwa dalam menyampaikan informasi kepada alumni masih belum menggunakan bantuan media apapun. Metode pengembangan sistem yang digunakan adalah *RDF*. Hasil dari penelitian berupa website sebagai media informasi alumni.

Penelitian lainnya yang berkaitan dengan website alumni ini diteliti oleh Faisal Satria Nur Ramadhan, Nur Chasanah, Dadang Iskandar pada tahun 2020 yang berjudul “Aplikasi Forum Diskusi Alumni Mahasiswa Teknik Universitas Jenderal Soedirman Menggunakan *Laravel*”. Penelitian ini menjelaskan bahwa dalam menyampaikan informasi kepada alumni masih belum menggunakan bantuan media apapun. Metode pengembangan sistem yang digunakan adalah *UML*. Hasil dari penelitian berupa website sebagai media informasi alumni.

Berdasarkan tinjauan pustaka tersebut terdapat beberapa persamaan dengan website alumni yang akan dibangun yang terletak pada bahasa pemrograman dan *database* yang digunakan, sedangkan perbedaan dengan website yang akan dibangun pengguna dapat melakukan diskusi atau tanya jawab dengan topik pembahasan tertentu. Pada penelitian ini akan melakukan penyempurnaan dalam penyampaian informasi website dengan berbasis forum.

2.2. Kerangka Pemikiran

Kebutuhan website alumni bagi prodi informatika Universitas Sahid Surakarta dapat di bentuk diagram kerangka pemikiran yang dapat dilihat seperti Gambar 2.1 dibawah ini.



Gambar 2.1 Kerangka Pemikiran

Pada Gambar 2.1 di atas dapat dijelaskan bahwa yang menjadi Latar Belakang yaitu Prodi Informatika pada Universitas Sahid Surakarta, lalu mengidentifikasi masalah yang yaitu Forum Diskusi Alumni, lalu mengumpulkan Data Alumni, lalu menganalisa melalui 3(tiga) tahapan yaitu Studi Pustaka, Wawancara, serta *Observasi*(Pengamatan), lalu tersimpulkan untuk penyelesaian masalahnya itu Pembuatan Website, dan Hasilnya berupa Website Alumni Informatika.

2.3. Teori Pendukung

2.3.1. Diagram UML

UML adalah sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis objek. *UML* merupakan singkatan dari *Unified Modeling Language*. *UML* juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan. Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat pengembangan karena developer harus melakukan penelusuran dan mempelajari kode program. *UML* diagram adalah diagram yang didasarkan pada *Unified Modelling Language (UML)* yaitu bahasa pemodelan visual standar yang digunakan untuk pemodelan bisnis dan proses serupa, analisis, desain, dan implementasi sistem berbasis perangkat lunak. Dalam *UML* sendiri terdapat beberapa diagram yang wajib dikuasai yaitu:


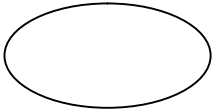


a. *Structural* Diagram

1. *Class Diagram*, diagram ini terdiri dari *class*, *interface*, *association*, dan *collaboration*. Diagram ini menggambarkan objek - objek yang ada di sistem.
2. *Object Diagram*, diagram ini menggambarkan hasil instansi dari *class* diagram. Diagram ini digunakan untuk membuat *prototype*.
3. *Component Diagram*, diagram ini menggambarkan kumpulan komponen dan hubungan antar komponen. Komponen terdiri dari *class*, *interface*, atau *collaboration*
4. *Deployment Diagram*, diagram ini menggambarkan kumpulan *node* dan hubungan antar *node*. *Node* adalah entitas fisik dimana komponen di *deploy*. Entitas fisik ini dapat berupa server atau perangkat keras lainnya.



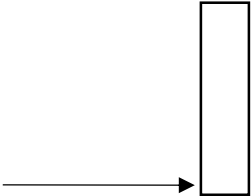
b. *Behavioral Diagram*

1. *Usecase Diagram*, diagram ini menggambarkan kumpulan *usecase*, aktor, dan hubungan mereka. *Usecase* adalah hubungan antara fungsionalitas sistem dengan aktor internal/eksternal dari sistem.
2. *Sequence Diagram*, diagram ini menggambarkan interaksi yang menjelaskan bagaimana pesan mengalir dari objek ke objek lainnya.
3. *Collaboration Diagram*, diagram ini merupakan bentuk lain dari *sequence diagram*. Diagram ini menggambarkan struktur organisasi dari sistem dengan pesan yang diterima dan dikirim.
4. *Statechart Diagram*, diagram ini menggambarkan bagaimana sistem dapat bereaksi terhadap suatu kejadian dari dalam atau luar. Kejadian(*event*) ini bertanggung jawab terhadap perubahan keadaan sistem.
5. *Activity Diagram*, menggambarkan aliran kontrol sistem. Diagram ini digunakan untuk melihat bagaimana sistem bekerja ketika dieksekusi. (Fajar, 2016)

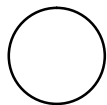
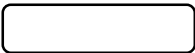
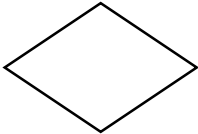
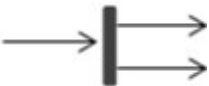

Tabel 2.1 *Usecase* Diagram

Usecase			
No	Gambar	Nama	Keterangan
1		Actor	Orang proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari <i>actor</i> adalah gambar orang, biasanya dinyatakan menggunakan kata benda diawal frase nama <i>actor</i> .
2		Usecase	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau <i>actor</i> biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>usecase</i> .
3		Association	Komunikasi antara <i>actor</i> dan <i>usecase</i> yang berpartisipasi pada <i>usecase</i> atau <i>usecase</i> memiliki interaksi dengan <i>actor</i> .
4		Generalization	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).




Tabel 2.2 *Sequence Diagram*

Sequence			
No	Gambar	Nama	Keterangan
1		Actor	Orang proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari <i>actor</i> adalah gambar orang, biasanya dinyatakan menggunakan kata benda di awal frase nama <i>actor</i> .
2		Life Line	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
3		Message	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi.



Tabel 2.3 *Activity Diagram*

Activity			
No	Gambar	Nama	Keterangan
1		Initial	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2		Activity	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3		Decision	Asosiasi percabangan dimana lebih dari satu aktivitas digabungkan menjadi satu.
4		Join	Asosiasi penggabungan dimana lebih dari satu aktivitas lebih dari satu.
5		Final	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status satu.



Tabel 2.4 *Class Diagram*

Class			
No	Gambar	Nama	Keterangan
1		Class	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
2		Asosiasi	Komunikasi antara <i>actor</i> dan <i>usecase</i> yang berpartisipasi pada <i>usecase</i> atau <i>usecase</i> memiliki interaksi dengan <i>actor</i> .
3		Generalisasi	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).

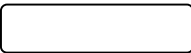

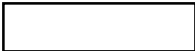
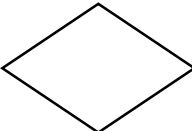


Tabel 2.5 *Component Diagram*

Component			
No	Gambar	Nama	Keterangan
1		Component	Komponen Sistem.
2		Link	Menggambarkan relasi antar <i>node</i> .

Tabel 2.6 *Deployment Diagram*

Deployment			
No	Gambar	Nama	Keterangan
1		Node	Mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika di dalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang telah didefinisikan sebelumnya pada diagram komponen.
2		Link	Menggambarkan relasi antar <i>node</i> .

Tabel 2.7 *Flow Chart* Diagram

Flow Chart			
No	Gambar	Nama	Keterangan
1		Start / Finish	Digunakan untuk menggambarkan awal dan akhir dari suatu kegiatan.
2		Input / Output	Digunakan untuk menggambarkan proses masukan data yang berupa pembicaraan data dan sekaligus proses keluaran yang berupa pencetakan data.
3		Proses	Digunakan untuk menggambarkan proses yang sedang di eksekusi.
4		Decision	Digunakan untuk menggambarkan proses pengujian suatu kondisi yang ada.
5		Data Base	Tempat penyimpanan suatu data.
6		Line	Digunakan untuk menggambarkan hubungan proses dari suatu proses ke proses lainnya.

2.3.2. Aplikasi

Aplikasi merupakan penerapan, menyimpan sesuatu hal, data, permasalahan, pekerjaan ke dalam suatu sarana atau media yang dapat digunakan untuk diterapkan menjadi sebuah bentuk yang baru. (Jogiyanto HM, 2017)

Pengertian aplikasi secara umum adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya aplikasi merupakan suatu perangkat komputer yang siap pakai bagi *user*.

2.3.3. Forum

Forum menurut KBBI merupakan tempat atau wadah untuk membicarakan kepentingan bersama atau tempat pertemuan dan bertukar pikiran antar pengguna. Forum ini merupakan tempat pengguna untuk mencari informasi yang relevan dan memberikan masukan masukkan sesama antar pengguna.

Diskusi adalah suatu proses yang teratur yang sudah melibatkan sekelompok orang untuk saling berinteraksi, bisa berasal dengan pengalaman, informasi, kesimpulan, serta solusi untuk sebuah masalah yang didiskusikan.

Forum diskusi adalah, “Proses, pengetahuan, pembelajaran untuk memberikan arti internalisasi nilai - nilai kepada peserta diskusi atau melihat keadaan diskusi pada forum tersebut. (Fisal, 2020)

2.3.4. Website

Sebuah situs *web* biasanya ditempatkan setidaknya pada sebuah *server web* yang dapat diakses melalui jaringan seperti Internet, ataupun jaringan area lokal (*LAN*) melalui alamat Internet yang dikenali sebagai *URL*. Gabungan atas semua situs yang dapat diakses publik di Internet disebut pula sebagai *World Wide Web* atau lebih dikenal dengan singkatan *WWW*.

Meskipun setidaknya halaman beranda situs Internet umumnya dapat diakses publik secara bebas, pada praktiknya tidak semua situs memberikan kebebasan bagi publik untuk mengaksesnya, beberapa situs *web* mewajibkan pengunjung untuk melakukan pendaftaran sebagai anggota, atau bahkan meminta pembayaran untuk dapat menjadi anggota untuk dapat mengakses isi yang terdapat dalam situs *web* tersebut, misalnya situs-situs yang menampilkan pornografi, situs-situs berita, layanan surel (*e-mail*), dan lain-lain. Pembatasan-pembatasan ini umumnya dilakukan karena alasan keamanan, menghormati privasi, atau karena tujuan komersial tertentu.

Halaman-halaman *web* tersebut diakses oleh pengguna melalui protokol komunikasi jaringan yang disebut sebagai *HTTP*, sebagai tambahan untuk meningkatkan aspek keamanan dan aspek privasi yang lebih baik, situs *web* dapat pula mengimplementasikan mekanisme pengaksesan melalui protokol *HTTPS*. (Swara, 2016)

2.3.5. HTML

HTML atau *Hypertext Markup Language* merupakan salah satu bahasa yang biasa digunakan oleh pengguna dalam membuat tampilan yang digunakan oleh *web application*.(Endar,2018)

2.3.6. PHP

PHP adalah bahasa pemrograman yang sering disisipkan ke dalam *HTML*. *PHP* sendiri berasal dari kata *Hypertext Preprocessor*. Sejarah *PHP* pada awalnya merupakan kependekan dari *Personal Home Page* (Situs personal). *PHP* pertama kali dibuat oleh *Rasmus Lerdorf* pada tahun 1995. Pada waktu itu *PHP* masih bernama *Form Interpreted (FI)*, yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari *web*.

Bahasa pemrograman ini menggunakan sistem *server-side*. *Server-side programming* adalah jenis bahasa pemrograman yang nantinya *script/program* tersebut akan dijalankan/diproses oleh *server*. Kelebihannya adalah mudah digunakan, sederhana, dan mudah untuk dimengerti dan dipelajari.

Semenjak *PHP* menjadi bahasa pemrograman yang *open source*, pengembang tidak *Perlu* menunggu sampai dengan *update* terbaru rilis.

Pengguna *PHP* akan lebih baik jika menggunakan versi terbaru. Sehingga jika ada rilis terbaru Anda harus menyesuaikan sistem Anda dengan versi *PHP* yang paling baru. Meskipun harus menggunakan versi terbaru, biaya untuk *maintenance* dan *web development* sangat terjangkau.

Bahasa pemrograman *PHP* membantu Anda untuk mengembangkan aplikasi berbasis *web* yang cukup kompleks, handal, dan cepat. Tergantung dari spesifikasi bisnis, penggunaan *hosting*, tingkat pengalaman, kebutuhan aplikasi, dan pengembangan *timeframe*. Selain itu ada banyak *PHP frameworks* yang dapat Anda pilih. (Yasin, 2019)

2.3.7. CSS/SCSS/SASS

CSS berasal dari kata *Cascading Style Sheet*. CSS adalah bahasa pemrograman yang digunakan untuk mengatur style di dalam website atau bisa disebut dengan kumpulan kode-kode yang bertujuan untuk mengatur tampilan/layout halaman web supaya terlihat lebih menarik oleh pengunjung.

Web programmer dan blogger menggunakan CSS untuk menentukan tata letak, warna, *background*, jenis *font*, ukuran, dan hampir semua hal yang berhubungan dengan tampilan website. Ini yang membuat CSS hampir ada di dalam banyak website di internet.

CSS biasanya diletakkan pada *file* terpisah. Sehingga jika ada fungsi atau skrip yang ingin menggunakan CSS tinggal memanggilnya saja. Langkah ini akan menyederhanakan skrip HTML yang cukup panjang jika ingin merubah warna, *font*, dan lain sebagainya.

SASS (*SASSY Stylesheet*) sendiri memberi kemudahan seperti tidak usah menulis kurung buka /tutup, dan juga tidak Perlu menulis titik koma diakhir *sintaks*. Dengan menggunakan SASS sendiri memudahkan kita dalam menulis CSS seperti penggunaan *variabel*, *nesting*, *mixins*, *selector inheritance* dll. Keunggulan lainnya seperti CSS yang lebih terstruktur, rapi, mudah dipahami, dan yang paling penting dapat berjalan baik di semua browser.

Lalu ada SCSS, sama seperti SASS namun mungkin bagi kalian yang sedikit kurang paham dengan SASS bisa menggunakan SCSS, SCSS merupakan *sintaks* yang paling umum digunakan yang merupakan *superset* dari CSS, yang berarti setiap sintak CSS yang berada di CSS3 bisa digunakan pada SCSS, tetapi bisa menggunakan fitur seperti yang ada pada SASS dan memang penulisannya lebih mudah dipahami bagi kalian yang baru mengenal SASS dan SCSS ini. Biasanya *file* SCSS menggunakan format ".SCSS". (Yasin, 2019)

2.3.8. Java Script

JavaScript merupakan bahasa pemrograman yang menjadikan website Anda lebih hidup dan menarik. Bahasa ini berbeda dari *HTML* (yang mengatur konten) dan *CSS* (yang mengelola *layout*). Berbeda dari *PHP*, bahasa pemrograman ini dijalankan di perangkat pengunjung situs Anda dan bukannya di *server*. (Arita, 2019)

2.3.9. Composer

Composer adalah *tools dependency manager* pada *PHP*, *Dependency* (ketergantungan) sendiri diartikan ketika *project PHP* yang kamu kerjakan masih membutuhkan atau memerlukan *library* dari luar. *Composer* berfungsi sebagai penghubung antara *project PHP* kamu dengan *library* dari luar.

Jika Bahasa pemrograman *PHP* menggunakan *Composer* sebagai *dependency manager*, Maka sama halnya seperti Ruby yang menggunakan *Gem*, Java menggunakan *Maven* and *Gradle* dan seluruh komunitas JS berfokus pada *npm*. (Beon, 2020)

2.3.10. MYSQL

MYSQL adalah salah satu sistem manajemen basis data *SQL* (*Structured Query Language*). *MYSQL* menjadi salah satu *database* pertama yang bersifat *open source*. Pengguna yang masih banyak membuatnya bertahan sampai dengan saat ini.

Sistem manajemen ini dibuat dan dikembangkan pertengahan tahun 90-an. Sampai dengan saat ini masih ada beberapa sistem manajemen *database* selain *MYSQL* yang banyak digunakan oleh pengguna maupun pengembang sistem berbasis *web*. (Arita, 2019)

2.3.11. DATA BASE

Dalam bahasa yang sederhana, *database* adalah sekumpulan data yang terstruktur. Anggap saja Anda sedang berpose *selfie*: yang dilakukan pastilah menekan tombol kamera depan, lalu mengabadikan potret diri. Foto *selfie* yang diambil adalah data, sedangkan galeri foto merupakan *database*. *Database* adalah tempat untuk menyimpan dan mengelola data. Kata “relasional” berarti data yang disimpan di *dataset* dikelola sebagai tabel. Setiap tabel saling terkait. Jika *software* tidak mendukung model data relasional, maka yang dipanggil hanyalah DBMS. (Arita, 2019)

2.3.12. VSCODE

Visual Studio Code adalah editor kode sumber gratis yang dibuat oleh *Microsoft* untuk *Windows*, *Linux*, dan *macOS*. Fitur termasuk dukungan untuk *debugging*, *syntax highlighting*, *intelligent code complete*, *snippet*, *code refactoring*, dan *embedded Git*. Pengguna dapat mengubah tema , pintasan *keyboard*, *preferences*, dan memasang ekstensi yang menambahkan fungsionalitas tambahan.

2.3.13. XAMPP

Pengertian *XAMPP* adalah perangkat lunak (*software*) bebas, yang mendukung untuk banyak sistem operasi, yang merupakan kompilasi dari beberapa program. Fungsi *XAMPP* sendiri adalah sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri beberapa program antara lain : *Apache HTTP Server*, *MYSQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*.

Nama *XAMPP* sendiri merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MYSQL*, *PHP* dan *Perl*. Masing-masing huruf yang ada di dalam nama *XAMPP* memiliki arti sebagai berikut ini:

1. X = *Cross Platform*

Merupakan kode penanda untuk *software cross platform* atau yang bisa berjalan di banyak sistem operasi.

2. A = *Apache*

Apache adalah aplikasi *web server* yang bersifat gratis dan bisa dikembangkan oleh banyak orang (*open source*).

3. M = *MYSQL / MariaDB*

MYSQL atau *MariaDB* merupakan aplikasi *database server* yang dikembangkan oleh orang yang sama. *MYSQL* berperan dalam mengolah, mengedit, dan menghapus daftar melalui *database*.

4. P = *PHP*

Huruf “P” yang pertama dari akronim kata *XAMPP* adalah inisial untuk menunjukkan eksistensi bahasa pemrograman *PHP*. Bahasa pemrograman ini biasanya digunakan untuk membuat website dinamis, contohnya dalam website berbasis *CMS WordPress*.

5. P = *Perl*

Sementara itu, untuk huruf P selanjutnya merupakan singkatan dari bahasa pemrograman *Perl* yang kerap digunakan untuk memenuhi berbagai macam kebutuhan. *Perl* ini bisa berjalan di dalam banyak sistem operasi sehingga sangat fleksibel dan banyak digunakan. (Mufid, 2019)

2.3.14. Blackbox Testing

Blackbox Testing yaitu menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.(Rosa,2016)

Pengujian *Blackbox* adalah pengujian aspek fundamental *system* tanpa memperhatikan struktur logika internal perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar, serta data yang diuji didasarkan pada spesifikasi perangkat lunak, dimana apabila dieksekusi pada perangkat lunak dan kemudian keluaran dari perangkat lunak dicek telah sesuai yang diharapkan. Dalam metode ini berusaha untuk menemukan kesalahan beberapa kategori :

1. Fungsi – fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses *datebase*.

2.3.15. Webqual Testing

Webqual berdasar pada konsep *Quality Function Deployment* (QFD) adalah suatu pengukuran untuk mengukur kualitas dari sebuah website berdasarkan instrumen yang dapat dikategorikan kedalam tiga variabel yaitu: *usability*, *information quality*, dan *interaction quality*. Pada ketiga variabel bertujuan untuk mengukur kepuasan konsumen atau pengguna terhadap kualitas website. Metode penelitian ini mulai dikembangkan sejak tahun 1998, Metode ini juga mengalami perubahan dalam penyusunan dimensi dan butir pertanyaan. Versi terbaru *Webqual* yaitu 4.0, *Webqual* 4.0 disusun berdasarkan tiga dimensi, yaitu: kualitas kegunaan (*usability*), kualitas informasi (*information quality*), dan kualitas interaksi (*interaction quality*) (Monalisa dan Rizky, 2021).

Webqual disusun berdasarkan penelitian ini terdiri atas tiga dimensi yaitu:

1. *Usability* (Kegunaan) *Usability* adalah mutu yang terkait dengan website, tampilan sebuah website merupakan faktor utama dalam mempengaruhi pengunjung website. Dalam hal ini, bertujuan membuat pengunjung atau pengguna merasa senang untuk mengunjungi website.
2. *Information Quality* (Kualitas Informasi) *Information Quality* adalah mutu dari isi yang terdapat pada website, dilihat dari isi pada web, berdasarkan ada atau tidak informasi yang disajikan untuk tujuan pengguna seperti akurasi, format dan keterkaitannya.

Interaction Quality (Kualitas Interaksi) *Interaction Quality* adalah mutu dari interaksi pelayanan yang dialami oleh pengguna mengakses ke dalam website yang terwujud dengan kepercayaan dan empati