

BAB 2

LANDASAN TEORI

2.1 Rapid Prototyping

Boar. B mengklaim bahwa Rapid Prototyping adalah "strategi khusus untuk melakukan pendefinisian persyaratan dimana kebutuhan pengguna diekstraksi, disajikan, dan berhasil disempurnakan dengan membangun sebuah model sistem dengan cepat"[1].

Rapid Prototyping selalu dimulai dengan membuat sketsa antarmuka. tujuannya bukan untuk mendapatkan tata letak aplikasi secara tepat, melainkan untuk memvisualisasikan fungsi aplikasi dan semua tugas yang mungkin dapat dilakukan.

Metodologi pembuatan antarmuka bertujuan untuk pengujian dimana menawarkan sebuah cara untuk memahami kebutuhan pengguna secara tepat. Oleh karena itu Rapid Prototyping merupakan metodologi perulangan dari pendekatan User Centered Design, dimana antarmuka dibuat, dievaluasi, dan disempurnakan sampai kriteria yang diinginkan tercapai[4].

2.2 Teknologi Website

World Wide Web (WWW) merupakan suatu sistem penyedia informasi dengan skala besar yang mengelola informasi tersebut secara terdistribusi dalam internet dengan menggunakan teknologi hypermedia. WWW mempunyai karakteristik sebagai berikut:

1. Dukungan interface yang terintegrasi WWW menyediakan suatu interface yang terintegrasi terhadap protokol, format data, sistem pengalamatan, dan lain-lain. Hal ini menyebabkan berbagai macam layanan dan basis data yang ada di internet dapat diakses secara langsung.
2. Kemudahan pada sisi pengguna WWW mendukung secara transparan sebagian besar aplikasi-aplikasi yang terdapat di internet seperti telnet, anonymous file transfer protocol, FTP, dan aplikasi lainnya.

3. Kemudahan dalam perkembangan Kapabilitas server WWW dapat dikembangkan secara mudah dengan menggunakan standar PHP antara server WWW dengan aplikasi yang lain. Program PHP juga memungkinkan perubahan informasi secara dinamis yang dapat diperoleh secara realtime.
4. Tidak tergantung pada platform tertentu WWW memungkinkan seseorang dapat membangun server WWW di berbagai sistem yang berlainan dan memberikan informasi dalam bentuk hypermedia. WWW tidak didesain untuk mendukung sistem tertentu.

WWW menggunakan model client server. WWW juga menggunakan ekspresi informasi, transfer informasi, metode penamaan informasi yang standar sehingga dapat memproses dan mentransfer informasi secara terdistribusi yang dilakukan secara sistematis.

Informasi yang terdistribusi disimpan di dalam server WWW dan user mengakses informasi tersebut menggunakan suatu software yang disebut dengan browser. Server menyimpan informasi di dalam file-file terstruktur menggunakan Hypertext Markup Language (HTML).

WWW menggunakan standar untuk mencapai konsistensi dalam menghasilkan dan melakukan transfer informasi. WWW menggunakan HTML sebagai standar produksi informasi dengan menggunakan Hypertext Transfer Protocol (HTTP) sebagai standar transfer informasi. WWW menggunakan standar penamaan yang disebut dengan Universal Resource Locator (URL) untuk mendukung penyediaan informasi secara universal dalam server yang terdistribusi.[5]

2.3 HTML Drag and Drop API

Drag and Drop HTML API memungkinkan aplikasi untuk menggunakan fitur *drag-and-drop* di browser. Pengguna dapat memilih komponen yang dapat digeser dengan mouse dan menggeser komponen tersebut ke komponen lain yang dapat menampung komponen tersebut, dan menjatuhkannya dengan melepaskan tombol mouse. Komponen yang sedang digeser mengikuti pointer selama operasi penggeseran. *Drag and Drop HTML API* menggunakan *DOM event model* dan *drag event* yang diwariskan dari *mouse event*. Operasi drag dimulai ketika pengguna

memilih komponen yang dapat digeser, menggeser komponen ke komponen yang dapat dijatuhkan, dan kemudian melepaskan komponen yang digeser.

Objek *DataTransfer* termasuk keadaan peristiwa menggeser, seperti jenis penggeseran yang dilakukan (seperti menyalin atau memindahkan), drag item (satu atau lebih item), dan tipe MIME (*Multipurpose Internet Mail Extension*) dari setiap drag item. Objek *DataTransfer* juga memiliki metode untuk menambah atau menghapus item ke *dragData*.

Antarmuka *DragEvent* dan *DataTransfer* harus menjadi satu-satunya yang diperlukan untuk menambahkan kemampuan HTML Drag and Drop ke suatu aplikasi. Setiap objek *DataTransfer* berisi properti item, yang merupakan daftar objek *DataTransferItem*. Objek *DataTransferItem* mewakili item penggeseran tunggal, masing-masing dengan properti jenis (baik string atau file) dan properti tipe untuk tipe MIME item data. Objek *DataTransferItem* juga memiliki metode untuk mendapatkan *drag data* item.

Objek *DataTransferItemList* adalah kumpulan objek *DataTransferItem*. *DataTransferItem* memiliki metode untuk menambahkan drag item ke list, menghapus drag item dari list, dan menghapus daftar semua drag item.

Perbedaan utama antara antarmuka *DataTransfer* dan *DataTransferItem* adalah bahwa yang pertama menggunakan metode *getData()* synchronous untuk mengakses *drag data* item, tetapi yang terakhir menggunakan metode *getAsString()* asynchronous[4].

2.4 Web Service

2.4.1 Definisi Web Service

Web service adalah sistem perangkat lunak yang dirancang untuk mendukung interaksi dari sistem ke sistem lain yang interoperable melalui suatu jaringan. *Web Service* memiliki arsitektur dalam format yang dapat diproses. Sistem lain berinteraksi dengan *Web Service* dengan cara yang ditentukan oleh deskripsinya menggunakan pesan SOAP, biasanya disampaikan menggunakan HTTP dengan pengkodean XML dalam hubungannya dengan standar terkait Web lainnya [7].

Web service bertujuan untuk meningkatkan kolaborasi antar pemrogram dan perusahaan, yang memungkinkan sebuah fungsi di dalam *Web Service* dapat dipinjam oleh aplikasi lain tanpa perlu mengetahui detail pemrograman yang terdapat di dalamnya.

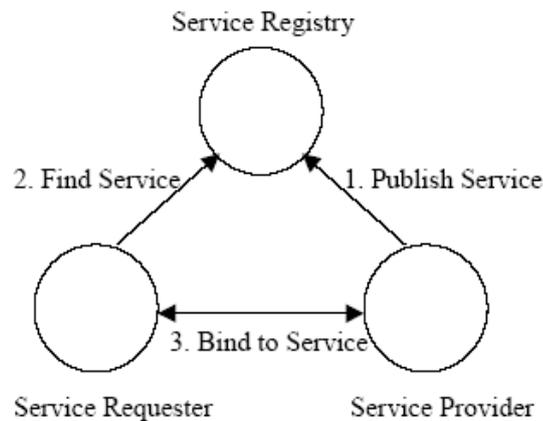
Beberapa alasan mengapa digunakannya web service adalah sebagai berikut:

1. *Web service* dapat digunakan untuk mentransformasikan satu atau beberapa bisnis logic atau class dan objek yang terpisah dalam satu ruang lingkup yang menjadi satu, sehingga tingkat keamanan dapat ditangani dengan baik.
2. *Web service* memiliki kemudahan dalam proses deploymentnya, karena tidak memerlukan registrasi khusus ke dalam suatu sistem operasi. Web service cukup di unggah ke web server dan siap diakses oleh pihak-pihak yang telah diberikan otorisasi.
3. *Web service* berjalan di port 80 yang merupakan protokol standar HTTP dan dapat dikonfigurasi sesuai kebutuhan.

2.4.2 Arsitektur Web Service

Web service memiliki tiga entitas dalam arsitekturnya, yaitu:

1. *Service Provider*: Berfungsi untuk menyediakan layanan/service dan mengolah sebuah registry agar layanan-layanan tersebut dapat tersedia.
2. *Service Registry*: Berfungsi sebagai lokasi central yang mendeskripsikan semua layanan/service yang telah di register.
3. *Service Requestor*: Peminta layanan yang mencari dan menemukan layanan yang dibutuhkan serta menggunakan layanan tersebut.



Gambar 2.1 Arsitektur web service

2.5 ReactJS

2.5.1 Definisi ReactJS

ReactJS adalah sebuah pustaka/library javascript buatan Facebook yang bersifat open source untuk membangun User Interface [8]. ReactJS hanya *menghandle* semua hal yang berkaitan dengan tampilan dan logika di sekitarnya. ReactJS ini diciptakan dengan tujuan untuk membangun aplikasi skala besar dengan data yang berubah dan terus berubah dari waktu ke waktu.

2.5.2 Keuntungan Menggunakan ReactJS

1. Mudah dipahami

Gaya penulisan yang deklaratif membuat ReactJs mudah dipahami dan membuat react mudah diprediksi jika ada kesalahan penulisan kode.

2. JSX

JSX adalah sebuah ekstensi sintaks mirip dengan XML ke *ECMAScript* tanpa semantik yang spesifik. JSX tidak dimaksudkan untuk diterapkan oleh mesin atau browser. JSX bukan proposal untuk memasukkan JSX ke dalam spesifikasi *ECMAScript*. JSX dibuat untuk digunakan oleh berbagai *preprocessor* (transpiler) untuk mengubah token menjadi sebuah standar *ECMAScript* [9].

Tujuan pembuatan JSX adalah untuk mendefinisikan sintaksis ringkas dan akrab untuk mendefinisikan struktur pohon dengan atribut. Sintaksis generik

tetapi terdefinisi dengan baik memungkinkan komunitas parser independen dan stabil sintaksis untuk memenuhi spesifikasi tunggal.

Memasukkan sintaks baru dalam bahasa yang sudah ada adalah usaha berisiko. Implementator sintaksis lain atau bahasa yang ada dapat memperkenalkan ekstensi sintaksis lain yang tidak kompatibel.

JSX ini tidak berusaha untuk mematuhi spesifikasi XML atau HTML. JSX dirancang sebagai fitur *ECMAScript* dan kesamaan dengan XML hanya untuk keakraban. JSX sendiri digunakan sebagai sintaks dasar pada User Interface library seperti React.JS.

3. Modular

Untuk membuat aplikasi dengan skala besar, kita dapat menulis kode-kode dengan skala yang lebih kecil untuk di satukan menjadi aplikasi utuh, dan dapat digunakan kembali (reusable).

4. Scalable

React js dapat menangani dengan sangat baik sebuah program dengan skala yang besar yang dapat menampilkan perubahan data yang sangat kompleks.

5. Flexibel

Dengan belajar 1 library saja kita dapat membuat aplikasi Web, Mobile, maupun Desktop.

6. Efisien dan Cepat

ReactJS menciptakan Virtual DOM untuk mempercepat urusan perubahan DOM. Semua operasi di kerjakan di dalam Virtual DOM, setelah operasi selesai ReactJS menulis perubahan tersebut di dalam DOM. Contoh kasusnya seperti ini: “Jika kita menulis dalam secarik kertas menggunakan spidol, apabila terjadi kesalahan penulisan kita harus menulis di kertas yang baru. Berbeda jika kita menggunakan pensil, cukup menghapus dan memperbaiki pada bagian yang salah”.

7. Mudah untuk di Debugging

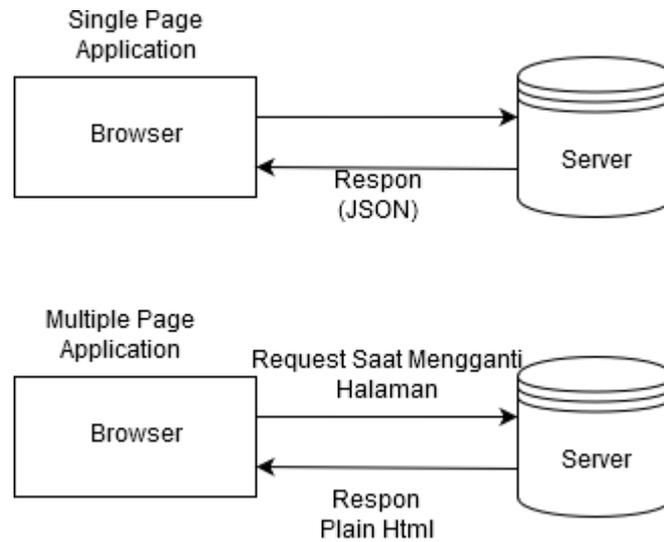
Ketika kita mulai menggunakan ReactJS, jangan lupa menginstall extensi resmi ReactJS. Kita dapat dengan mudah menjelajah Virtual DOM pada aplikasi yang sudah kita buat, sehingga jika ada bug bisa cepat ditemukan.

2.6 SPA

2.6.1 Definisi SPA

SPA (Single Page Application) merupakan aplikasi yang bekerja di dalam browser yang tidak membutuhkan *reload page* saat digunakan. Dengan kata lain, pengguna atau user tidak akan berpindah halaman dengan melakukan request kepada server setiap kali terjadi interaksi pada aplikasi. Yang membedakan SPA dengan *non-SPA* adalah *single page application* hanya akan melakukan *load* terhadap satu halaman dari *server* kemudian mekanisme *routing* yang biasanya di-*handle* oleh *server* kini dibebankan pada *client*. Akibatnya, website yang menggunakan SPA memiliki performa yang lebih cepat tanpa harus merequest kembali halaman secara terus menerus[10].

Single page application merupakan aplikasi yang menggunakan manfaat penuh di browser yang tidak membutuhkan pergantian halaman dengan cara merequest ke server. Sedangkan Multi Page application membutuhkan request ke server dan server mereturn berupa html ke browser. Artinya setiap perpindahan halaman pada Multi Page application membutuhkan request dari client ke server. Single page application biasanya melakukan request ke server selain mendapatkan initial file (js/html/css) juga mendapatkan data keluaran JSON(Javascript object notation) yang berfungsi sebagai data untuk komponennya. Untuk gambar Perbandingan website SPA & MPA dapat dilihat pada Gambar 2.2 dibawah ini :



Gambar 2.2 Perbandingan Website SPA & MPA

2.6.2 Keuntungan menggunakan SPA

Adapun 2 keuntungan utama yang didapat ketika menerapkan *single page application* yaitu mengurangi penggunaan *bandwidth* jaringan dan proses navigasi yang lebih cepat dibanding dengan *Multiple Page Application*. *Bandwidth* jaringan berkurang dikarenakan *SPA* ini banyak didukung dengan *library JavaScript*. Sedangkan proses navigasi yang lebih cepat dikarenakan data yang diterima dari server berupa *JSON (JavaScript Object Notation)* sehingga bisa di-render secara *asynchronous* dengan menggunakan *JavaScript*. Beberapa *framework JavaScript* yang memakai konsep *SPA* diantaranya *React.js*, *Angular.js* dan *Vue.js*. Adapun beberapa contoh website *Single Page Application* diantaranya adalah Tokopedia, Facebook, Techinasia dan lain sebagainya.

2.7 PHP Programming Language

2.7.1 Definisi PHP

PHP adalah bahasa pemrograman server-side yang bersifat open source. PHP merupakan bahasa pemrograman yang bersifat dinamis karena halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh user. Mekanisme ini menyebabkan informasi yang diterima user selalu yang terbaru (up to date). PHP merupakan script yang dapat terintegrasi dengan HTML (Hypertext Markup Language) [11].

2.7.2 Keuntungan Menggunakan PHP

Berikut merupakan keuntungan dari menggunakan bahasa pemrograman PHP diantaranya :

1. Dapat membuat Web menjadi lebih Dinamis.
2. PHP sifatnya Open Source yang artinya bisa digunakan oleh siapa saja secara gratis.
3. Program yang dibuat dengan PHP dapat digunakan atau dijalankan oleh Semua Sistem Operasi karena PHP berjalan secara Web Base yang berarti semua Sistem Operasi bahkan HP yang memiliki Web Browser bisa menggunakan program PHP.
4. Aplikasi PHP lebih cepat dan mudah jika dibandingkan dengan ASP dan Java.
5. Mendukung beberapa paket Database seperti MySQL, Oracle, PostgreSQL, dan lain-lain.
6. Bahasa pemrograman PHP tidak membutuhkan Kompilasi / Compile dalam penggunaannya.
7. Banyak Web Server yang mendukung PHP seperti Apache, Lighttpd, IIS dan yang lainnya.
8. Dalam pengembangan Aplikasi PHP mudah karena banyak Dokumentasi, Referensi & Developer yang mendukung dalam pengembangannya.

2.8 CodeIgniter

2.8.1 Definisi CodeIgniter

CodeIgniter adalah Framework untuk pengembangan suatu Aplikasi, sebuah toolkit untuk orang yang ingin membangun situs web menggunakan PHP. Tujuannya adalah memungkinkan Developer untuk mengembangkan proyek lebih cepat dibandingkan menulis kode dari awal, dengan tersedianya serangkaian *library* yang kaya untuk tugas-tugas yang biasanya dibutuhkan, serta antarmuka yang sederhana dan struktur logis untuk mengakses *library*. CodeIgniter memungkinkan Developer secara kreatif fokus pada proyeknya dengan meminimalkan jumlah kode yang diperlukan untuk tugas yang diberikan[12].

2.8.2 Keuntungan Menggunakan CodeIgniter

Berikut merupakan beberapa dari keuntungan menggunakan CodeIgniter :

1. Performance luar biasa
2. Kompatibilitas yang luas dengan akun hosting standar yang menjalankan berbagai versi dan konfigurasi PHP.
3. Tidak banyak memerlukan konfigurasi
4. Tidak membutuhkan *PHP cli*
5. Tidak membatasi berdasarkan peraturan code.
6. Tidak dipaksa menggunakan *Templating Language*

2.9 Javascript Programing Language

JavaScript merupakan bahasa *script* populer yang dipakai untuk menciptakan halaman Web yang dapat berinteraksi dengan pengguna dan dapat merespon event yang terjadi pada halaman. JavaScript merupakan perekat yang menyatukan halaman-halaman Web.

JavaScript, awalnya dikenal sebagai LiveScript, dikembangkan oleh Brendan Eich di Netscape pada tahun 1995 yang menjadi bagian terintegrasi di dalam Netscape Navigator 2.0. JavaScript merupakan bahasa *script* yang menghidupkan halaman-halaman HTML. JavaScript dapat dijalankan pada hampir semua platform. JavaScript merupakan bahasa sisi-klien yang didesain pada browser komputer Anda, bukan pada server. Ia dibangun secara langsung ke dalam browser, Microsoft Internet Explorer, Mozilla Firefox, Google Chrome. Opera, hampir semua browser. Dalam sintaks JavaScript mirip dengan C, Perl, dan Java.

Karena JavaScript terkait dengan browser. ia sangat terintegrasi dengan HTML. Ketika browser memuat sebuah halaman, server akan mengirim konten utuh dari dokumen, termasuk HTML dan statemen - statemen JavaScript. Konten HTML kemudian dibaca dan diinterpretasi baris demi baris sampai tag pembuka JavaScript dibaca, pada saat itu interpreter JavaScript mengambil alih. Ketika tag penutup JavaScript diraih. pemrosesan HTML berlanjut.

Meskipun JavaScript secara umum digunakan sebagai bahasa *script* sisi - klien. ia dapat pula digunakan (pada konteks lain) selain pada browser Web. Netscape menciptakan JavaScript sisi - server yang bisa diprogram sebagai bahasa CGI seperti Python atau Perl. [13].

2.10 Database

Database atau basis data adalah kumpulan data yang disimpan secara sistematis di dalam komputer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur, dan juga batasan-batasan data yang akan disimpan. Basis data merupakan aspek yang sangat penting dalam sistem informasi dimana basis data merupakan gudang penyimpanan data yang akan diolah lebih lanjut. Basis data menjadi penting karena dapat menghindari duplikasi data, hubungan antar data yang tidak jelas, organisasi data, dan juga update yang rumit[14].

Proses memasukkan dan mengambil data ke dan dari media penyimpanan data memerlukan perangkat lunak yang disebut dengan sistem manajemen basis data (database management system | DBMS). DBMS merupakan sistem perangkat lunak yang memungkinkan user untuk memelihara, mengontrol, dan mengakses data secara praktis dan efisien. Dengan kata lain semua akses ke basis data akan ditangani oleh DBMS. Ada beberapa fungsi yang harus ditangani DBMS yaitu mengolah pendefinisian data, dapat menangani permintaan pemakai untuk mengakses data, memeriksa sekuriti dan integriti data yang didefinisikan oleh DBA (Database Administrator), menangani kegagalan dalam pengaksesan data yang disebabkan oleh kerusakan sistem maupun disk, dan menangani unjuk kerja semua fungsi secara efisien[14].

Tujuan utama dari DBMS adalah untuk memberikan tinjauan abstrak data kepada user (pengguna). Jadi sistem menyembunyikan informasi tentang bagaimana data disimpan, dipelihara, dan tetap dapat diambil (akses) secara efisien. Pertimbangan efisien disini adalah bagaimana merancang struktur data yang

kompleks tetapi masih tetap bisa digunakan oleh pengguna awam tanpa mengetahui kompleksitas strukturnya. Database dibagi menjadi dua jenis yaitu:

1. Basis Data Flat File

Basis data flat-file ideal untuk data berukuran kecil dan dapat dirubah dengan mudah. Pada dasarnya, mereka tersusun dari sekumpulan string dalam satu atau lebih file yang dapat diurai untuk mendapatkan informasi yang disimpan. Basis data flat-file baik digunakan untuk menyimpan daftar atau data yang sederhana dan dalam jumlah kecil. Basis data flat-file akan menjadi sangat rumit apabila digunakan untuk menyimpan data dengan struktur kompleks walaupun dimungkinkan pula untuk menyimpan data semacam itu. Salah satu masalah menggunakan basis data jenis ini adalah rentan pada korupsi data karena tidak adanya penguncian yang melekat ketika data digunakan atau dimodifikasi.

2. Basis Data Relasional

Basis data ini mempunyai struktur yang lebih logis terkait cara penyimpanan. Kata "relasional" berasal dari kenyataan bahwa tabel-tabel yang berada di basis data dapat dihubungkan satu dengan lainnya. Basis data relasional menggunakan sekumpulan tabel dua dimensi yang masing masing tabel tersusun atas baris (tuple) dan kolom (atribut). Untuk membuat hubungan antara dua atau lebih tabel, digunakan key (atribut kunci) yaitu primary key di salah satu tabel dan foreign key di tabel yang lain. Saat ini, basis data relasional menjadi pilihan karena keunggulannya. Beberapa kelemahan yang mungkin dirasakan untuk basis data jenis ini adalah implementasi yang lebih sulit untuk data dalam jumlah besar dengan tingkat kompleksitasnya yang tinggi dan proses pencarian informasi yang lebih lambat karena perlu menghubungkan tabel-tabel terlebih.

2.11 UML

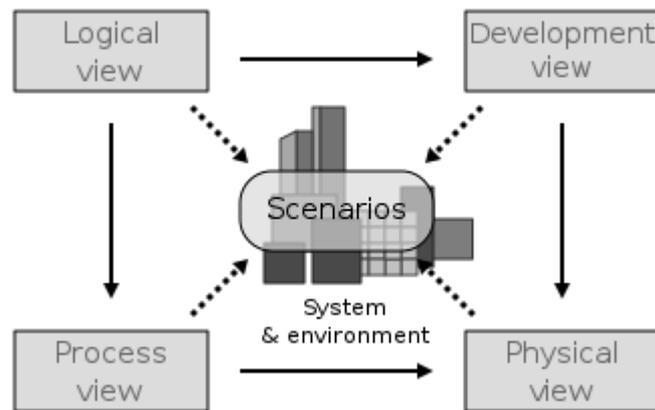
UML (Unified Modelling Language) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi objek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi

pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi Sharing) dan mengkomunikasikan rancangan mereka dengan yang lain[15].

UML adalah hasil kerja dari konsorsium berbagai organisasi yang berhasil dijadikan sebagai standar baku dalam OOAD (*Object Oriented Analysis Design*).

Sebagai sebuah notasi grafis yang relatif sudah dibakukan (open standard) dan dikontrol oleh OMG (Object Management Group - Mungkin lebih dikenal sebagai badan yang berhasil membakukan CORBA (*Common Object Request Broker Architecture*)). UML. menawarkan banyak keistimewaan UML tidak hanya dominan dalam penotasian di lingkungan OO tetapi juga populer di luar lingkungan OO. Paling tidak ada tiga karakter penting yang melekat di UML yaitu sketsa, cetak biru dan bahasa pemrograman. Sebagai sebuah sketsa, UML bisa berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dari sistem. Dengan demikian semua anggota tim akan mempunyai gambaran yang sama tentang suatu sistem. UML bisa juga berfungsi sebagai sebuah cetak biru karena sangat lengkap dan detail. Dengan cetak biru ini maka akan bisa diketahui informasi detail tentang coding program (forward engineering) atau bahkan membaca program dan menginterpretasikannya kembali ke dalam diagram (reverse engineering). Reverse engineering sangat berguna pada situasi dimana code program yang tidak terdokumentasi akan dimodifikasi/ dipelihara. Hal ini bisa terjadi ketika dokumentasi asli hilang atau bahkan belum dibuat sama sekali. Sebagai bahasa pemrograman, UML dapat menerjemahkan diagram yang ada di UML menjadi code program yang siap untuk dijalankan.

UML dibangun atas model 4+1 view. Model ini didasarkan pada fakta bahwa struktur sebuah sistem dideskripsikan dalam 5 view dimana salah satu di antaranya scenario. Scenario ini memegang peran khusus untuk mengintegrasikan content ke view yang lain.



Gambar 2.3 Model 4+1 View

Kelima view tersebut berhubungan dengan diagram yang dideskripsikan di UML. Setiap view berhubungan dengan perspektif tertentu dimana sistem akan diuji. View yang berbeda akan menekankan pada aspek yang berbeda dari sistem yang mewakili ketertarikan kelompok stakeholder tertentu. Penjelasan lengkap tentang sistem bisa dibentuk dengan menggabungkan informasi-informasi yang ada pada kelima view tersebut.

Skenario menggambarkan interaksi antara objek dan di antara proses. Skenario ini digunakan untuk identifikasi elemen arsitektur, ilustrasi dan validasi desain arsitektur serta sebagai titik awal untuk pengujian prototipe arsitektur. Skenario ini biasa juga disebut dengan use case view. Use case view ini mendefinisikan kebutuhan sistem karena mengandung semua view yang lain yang mendeskripsikan aspek-aspek tertentu dari perancangan sistem. Itulah sebabnya use case view menjadi pusat peran dan sering dikatakan yang mendrive proses pengembangan perangkat lunak.

Development view menjelaskan sebuah sistem dari perspektif programmer dan terkonsentrasikan ke manajemen perangkat lunak. View ini dikenal juga sebagai implementation view. Diagram UML yang termasuk dalam development view di antaranya adalah component diagram dan package diagram.

Logical view terkait dengan fungsionalitas sistem yang dipersiapkan untuk pengguna akhir. Logical view mendeskripsikan struktur logika yang mendukung

fungsi-fungsi yang dibutuhkan di use case. Design view ini berisi object diagram, class diagram, state machine diagram dan composite structure diagram.

Physical view menggambarkan sistem dari perspektif sistem engineer. Fokus dari physical view adalah topologi sistem perangkat lunak. View ini dikenal juga sebagai deployment view. Yang termasuk dalam physical view ini adalah deployment diagram dan timing diagram.

Process view berhubungan erat dengan aspek dinamis dari sistem, proses yang terjadi di sistem dan bagaimana komunikasi yang terjadi di sistem serta tingkah laku sistem saat dijalankan. Process view menjelaskan apa itu concurrency, distribusi integrasi, kinerja dan lain-lain. Yang termasuk dalam process view adalah activity diagram, communication diagram, sequence diagram dan interaction overview diagram[15].

2.12 GrapesJS

2.12.1 Definisi GrapesJS

GrapesJS adalah Framework Web Builder open source, yang menggabungkan berbagai *tools* dan fitur dengan tujuan untuk membantu Pengguna untuk membangun template HTML tanpa pengetahuan tentang pengkodean[16].

2.12.2 Keunggulan GrapesJS

GrapesJS memiliki beberapa keunggulan diantaranya:

1. *Drag and Drop* Komponen

GrapesJS dibundle dengan serangkaian *block* (komponen) bawaan, dan juga dapat membuat *block* custom sendiri.

2. Style Tak Terbatas

GrapesJS mengimplementasikan modul Style Manager yang sederhana dan lengkap yang memungkinkan styling CSS dengan bebas dari setiap komponen di dalam kanvas, dan dapat di konfigurasi untuk menggunakan salah satu properti CSS.

3. Terstruktur dan Terkontrol

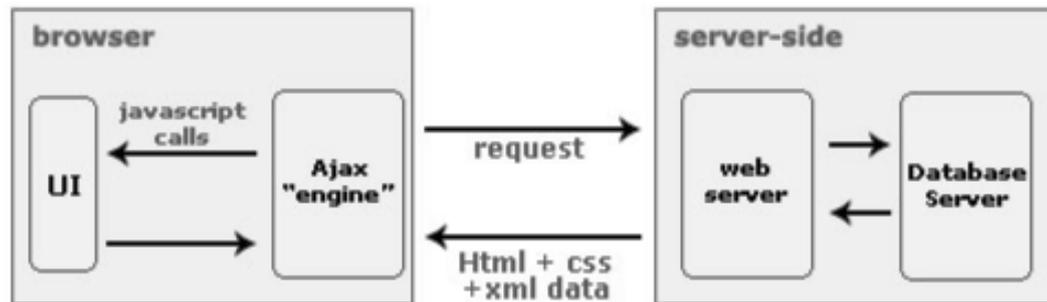
Dengan GrapesJS Anda dapat membuat *nested* komponen sebanyak banyaknya dan ketika struktur mulai menumpuk, *Layer Manager* menjadi

sangat berguna. Memungkinkan untuk mengelola dan mengatur ulang elemen dengan sangat cepat, selalu berfokus pada arsitektur struktur.

2.13 AJAX

AJAX adalah singkatan dari Asynchronous JavaScript and XML. AJAX merupakan teknik baru di dunia pengembangan halaman web yang berguna untuk membuat aplikasi web, yang lebih "kaya" dibandingkan dengan aplikasi web biasa. Apa triknya sehingga AJAX bisa membuat halaman web yang lebih kaya? Jawabannya adalah Ajax memungkinkan sebuah halaman web memperbarui data dari server yang ditampilkannya tanpa harus melakukan refresh, sehingga halaman terlihat lebih responsif.

Fitur dan ciri utama AJAX adalah sebuah halaman web tidak perlu di-refresh setiap kali ada data baru yang ingin ditampilkan. Dengan demikian, sebuah aplikasi web akan terasa seperti aplikasi desktop saja. Kecepatan, interaktivitas, fungsionalitas, dan tingkat kegunaan halaman web akan meningkat. Dari namanya, terlihat bahwa AJAX bersifat asynchronous. Arti asinkron di sini bahwa permintaan data tambahan dari server dan loading-nya dilakukan di background tanpa mempengaruhi tampilan dan sifat halaman web sekarang. Jadi, sembari AJAX bekerja untuk mengambil data dari server, user yang sedang membuka halaman web masih tetap bisa berinteraksi dengan halaman web tersebut, dimana halaman web tidak mengalami freezing seperti halnya aplikasi web konvensional. Sekarang mungkin Anda bertanya apakah AJAX itu sebuah bahasa pemrograman baru? Bukan, AJAX sebenarnya bukan bahasa pemrograman baru karena ternyata bahasa yang digunakan oleh AJAX adalah bahasa JavaScript yang sudah cukup berumur.



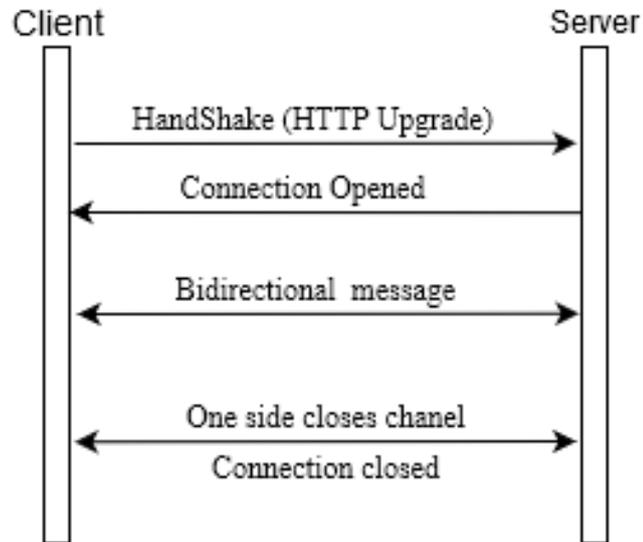
Gambar 2.4 Gambaran proses pengiriman data menggunakan AJAX

Data diambil dari server menggunakan objek XMLHttpRequest yang didukung berbagai browser modern. Dan untuk memformat konten yang didapat dari server tidak diperlukan persyaratan tambahan karena semua browser sudah mendukung DOM yang merupakan standar baku dari W3C. Oleh karena berbasis JavaScript, AJAX mewarisi segala keunggulan JavaScript. AJAX bersifat cross-platform dan bisa dijalankan di banyak browser, tanpa harus dibatasi oleh sistem operasi. AJAX juga bisa diaktifkan di berbagai komputer yang memiliki arsitektur yang berbeda-beda, dari mulai mainframe hingga desktop PC. Hal ini disebabkan teknologi yang digunakan untuk AJAX merupakan teknologi yang free dan open source, seperti JavaScript dan DOM. Selain itu, ada pula beragam implementasi framework yang free dan open source yang siap digunakan. Akibat arsitekturnya yang terbuka, implementasi AJAX bisa diterapkan di banyak bahasa pemrograman server-side, seperti PHP, ASP.NET, JSP, Ruby on Rails, dan sebagainya[17].

2.14 WebSocket

Protokol WebSocket memungkinkan komunikasi dua arah antara klien menjalankan sebuah kode dalam lingkungan yang terkendali kepada host yang telah ditargetkan atau dikonfigurasi untuk berkomunikasi secara realtime. Keamanan model yang digunakan dalam WebSocket adalah model keamanan yang umum digunakan oleh browser web. Protokol terdiri dari inisialisasi *handshake* diikuti oleh inisialisasi pesan dasar yang dilapisi TCP. Tujuan dari teknologi ini untuk menyediakan mekanisme aplikasi yang membutuhkan komunikasi dua arah dengan

server dan tidak bergantung pada pembukaan beberapa koneksi HTTP (misalnya menggunakan XMLHttpRequest atau <iframe> dan *long polling*) [18].



Gambar 2.5 Cara kerja websocket