**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

# **IDENTITAS PENGUSUL**

Nama : **Samsul Hadi**

NRP : **5107 100 152**

Dosen Wali : **Daniel O. Siahaan, S.Kom., M.Sc., PDEng.**

1. **JUDUL TUGAS AKHIR**

***“Perancangan dan Pembangunan Perangkat Lunak Sistem Monitoring Tugas Akhir Berbasis Incremental Rendering View Menggunakan Google Web Toolkit”***

1. **LATAR BELAKANG**

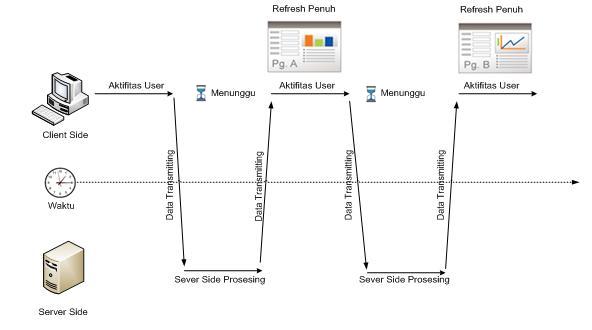
Berbagai aktifitas pendidikan di perguruan tinggi jelas memerlukan fasilitas monitoring atau administrasi baik administrasi akademik, kemahasiswaan, maupun keuangan. Perkembangan teknologi yang sangat pesat, terutama teknologi informasi semakin memungkinkan aktifitas adminitrasi dan monitoring pada aktifitas pendidikan menjadi lebih mudah dan efektif. Misalkan pada pelaksanaan administrasi antara mahasiswa, dosen, dan karyawan di Jurusan Teknik Informatika ITS Surabaya yang memanfaatkan teknologi internet untuk mempermudah proses tersebut. Dengan menggunakan teknologi internet, mahasiswa, karyawan, atau dosen dapat melakukan proses administrasi dan monitoring yang telah dilakukan itu secara online. Teknologi internet telah memberikan solusi yang mudah, murah, dan handal.

Perkembangan teknologi informasi pada masa sekarang yang sangat pesat menuntut keefektifan dalam segala hal termasuk proses monitoring pada aktifitas pendidikan, tak terkecuali monitoring tugas akhir. Dibutuhkan penerapan teknologi informasi agar proses monitoring menjadi lebih efektif. Dalam hal pengujian proposal tugas akhir, mahasiswa harus melalui beberapa prosedur. Proposal yang telah disetujui oleh dosen pembimbing akan diserahkan ke bagian koordinator tugas akhir sekaligus admin yang memiliki kegiatan-kegiatan yang berhubungan dengan penyusunan tugas akhir mahasiswa. Dengan menggunakan teknologi internet, mahasiswa dapat mengetahui status proposal yang telah diajukan apakah disetujui, direvisi atau ditolak.

Sampai saat ini penggunaan sistem monitoring tugas akhir di Jurusan Teknik Informatika ITS masih belum berfungsi secara maksimal. Hal ini terjadi karena banyak fungsionalitas dan struktur dari utilitas sistem yang belum bekerja dengan baik. Oleh karena itu, perlu dilakukan *reverse engineering* untuk meningkatkan kualitas dan memperbaiki pengelolaan dari perangkat lunak sistem monitoring tugas akhir tersebut.

Secara fundamental aplikasi sistem monitoring tugas akhir saat ini masih dikembangkan dalam model komunikasi web tradisional. Padahal, untuk saat ini pengembangan aplikasi web membutuhkan banyakkode sumber baik di sisi servermaupun sisi klien(JavaScript).

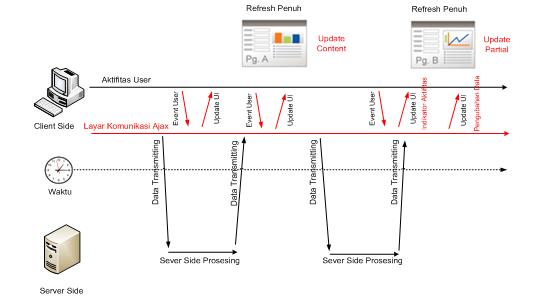
Web tradisional dikembangkan murni pada sisi server, sehingga seluruh aksi dari pengguna mengarah pada permintaan HTTP ke server. Kemudian server memperbaruhi *state* dan melakukan komputasi untuk halaman HTML baru dan mengirimnya ke klien. Pada metode tradisional tersebut kurang begitu menguntungkan bagi pengguna, Karena ketika dalam proses tersebut terjadi gangguan, maka browser akan memblokir secara serentak dan muncul *blank out* atau *blank window* sementara pengguna masih menunggu halaman baru.



**Gambar 1.** Model komunikasi web tradisional.

Metode pengembangan aplikasi web yang lebih modern untuk membangun sistem monitoring tugas akhir yaitu dengan menggunakan teknologi ajax yang berjalan pada sisi browser. Pada teknologi ini, aksi yang dilakukan pengguna mengarah pada browser yang menjalankan sebuah fungsi penanganan *event* JavaScript dengan membawa data dari halaman dan mengirimkan permintaan sebuah *asynchronous Xml Http Request* (XHR) dengan fungsi *response handler callback* tertentu.

*Incremental Rendering View* merupakan sebuah metode terbaru dari teknologi ajax yang digunakan untuk menangani kemampuan browser dalam menampilkan halaman web yang terunduh kepada pengguna secara parsial. Metode ini telah diterapkan di berbagai framework JavaScript yang menerapkan teknologi ajax seperti DOJO, Extjs, FORWARD, Google Web Toolkit, dan lain-lain. Metode *incremental rendering view* menangani proses pada lapisan data dan lapisan visual dari aplikasi. Pada lapisan data metode ini sangat bermanfaat untuk mengurangi adanya queri percabangan dan bersarang. Sedangkan pada lapisan visual, terjadi proses pembaharuan pada elemen DOM atau pemanggilan fungsi komponen JavaScript. Dengan metode ini, browser tidak lagi *blank out* dan tetap pada halaman yang lama selama permintaan sedang diproses. Pengguna bahkan bisa menambahkan aksi lainnya tanpa harus menunggu response dari permintaan yang dijalankan sebelumnya.

 **Gambar 2.** Model komunikasi ajax.

Google Web Toolkit (GWT) merupakan sebuah perangkat pengembangan untuk membangun aplikasi web berbasis ajax menggunakan bahasa Java, yang kemudian dikompile untuk mengoptimasi kode JavaScript. Pada hasil proses kompilasi ke JavaScript, ketika dijalankan browser akan merender menjadi HTML seperti sebuah halaman web tradisional tanpa memerlukan sebuah *plugin* atau bahkan JRE. Beberapa keunggulan dari proses kompilasi dari GWT yaitu untuk efisiensi, sekuritas, dan kompatibilitas browser.

Oleh karena itu, dalam Tugas Akhir ini penulis akan mengimplementasikan teknologi Google Web Toolkit. Sehingga sistem monitoring tugas akhir dapat dibangun menjadi aplikasi web berbasis *incremental rendering view* menggunakan ajax. Yang pada akhirnya diharapkan dapat tercipta sebuah sistem monitoring yang bermanfaat dan interaktif bagi pengguna.

1. **RUMUSAN MASALAH**

Berdasarkan latar belakang yang telah diuraikan diatas, terdapat beberapa permasalahan yang dapat dirumuskan sebagai berikut:

* + - 1. Bagaimana mengimplementasikan analisa proses bisnis untuk rekayasa perangkat lunak sistem monitoring tugas akhir dengan menggunakan pendekatan *reverse engineering* untuk menghasilkan perangkat lunak sistem monitoring tugas akhir yang baru.
      2. Bagamana perancangan dan pembangunan sistem monitoring tugas akhir dari hasil analisa proses bisnis untuk menghasilkan desain dan model pengembangan perangkat lunak.
      3. Bagaimana mengimplementasikan metode *Incremental Rendering View* pada halaman sistem monitoring tugas akhir dengan menggunakan Google Web Toolkit.

1. **BATASAN MASALAH**

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, diantaranya sebagai berikut:

* + - 1. Analisa proses bisnis dari sistem yang akan dibangun pada Tugas Akhir ini mengacu pada kasus prosedur pengerjaan tugas akhir di Jurusan Teknik Informatika ITS.
      2. Sistem monitoring yang akan dibangun berjalan secara *stand-alone* tanpa ada kolaborasi dari sistem lain, seperti sistem akademik, webmail, SKEM, dan sistem lainnya.
      3. Sistem monitoring yang akan dibangun menerapkan metode *incremental rendering view* dengan menggunakan teknologi ajax.
      4. Implementasi ajax pada sistem dibangun dan dikembangkan dalam bahasa Java dengan menggunakan framework Google Web Toolkit.
      5. Perangkat pengembangan Java (IDE) yang digunakan adalah Eclipse.
      6. Aplikasi yang dibangun bersifat *web-based* dengan menggunakan framework web Ruby on Rails dan sistem database yang digunakan yaitu MySQL.
      7. Aplikasi yang dibangun berjalan pada lingkungan browser yang mendukung JavaScript.

1. **TUJUAN TUGAS AKHIR**

Tujuan dari tugas akhir ini adalah sebagai berikut:

1. Merancang aplikasi Sistem Monitoring Tugas Akhir dengan metode *incremental rendering view* berbasis ajax dengan menggunakan framework Google Web Toolkit.
2. Memonitoring status proposal yang telah diajukan mahasiswa.
3. Memonitoring status tugas akhir dari proposal diterima hingga sidang tugas akhir.
4. Menampilkan rekapitulasi seluruh proposal tugas akhir dan judul tugas akhir sepanjang masa.
5. Menampilkan laporan masa dan progres tugas akhir mahasiswa.
6. Menampilkan informasi dan berita mengenai tugas akhir mahasiswa, tawaran topik tugas akhir, serta jadwal sidang proposal.
7. **MANFAAT TUGAS AKHIR**

Manfaat yang diharapkan dari tugas akhir ini adalah membuat suatu aplikasi online berbasis web yang interaktif dengan fungsi utilitas tinggi untuk membantu proses monitoring tugas akhir sehingga diharapkan dapat membantu pekerjaan dari koordinator tugas akhir, dosen pembimbing, dan dosen penguji serta mahasiswa dalam pelaksanaan tugas akhirnya.

1. **RINGKASAN TUGAS AKHIR**

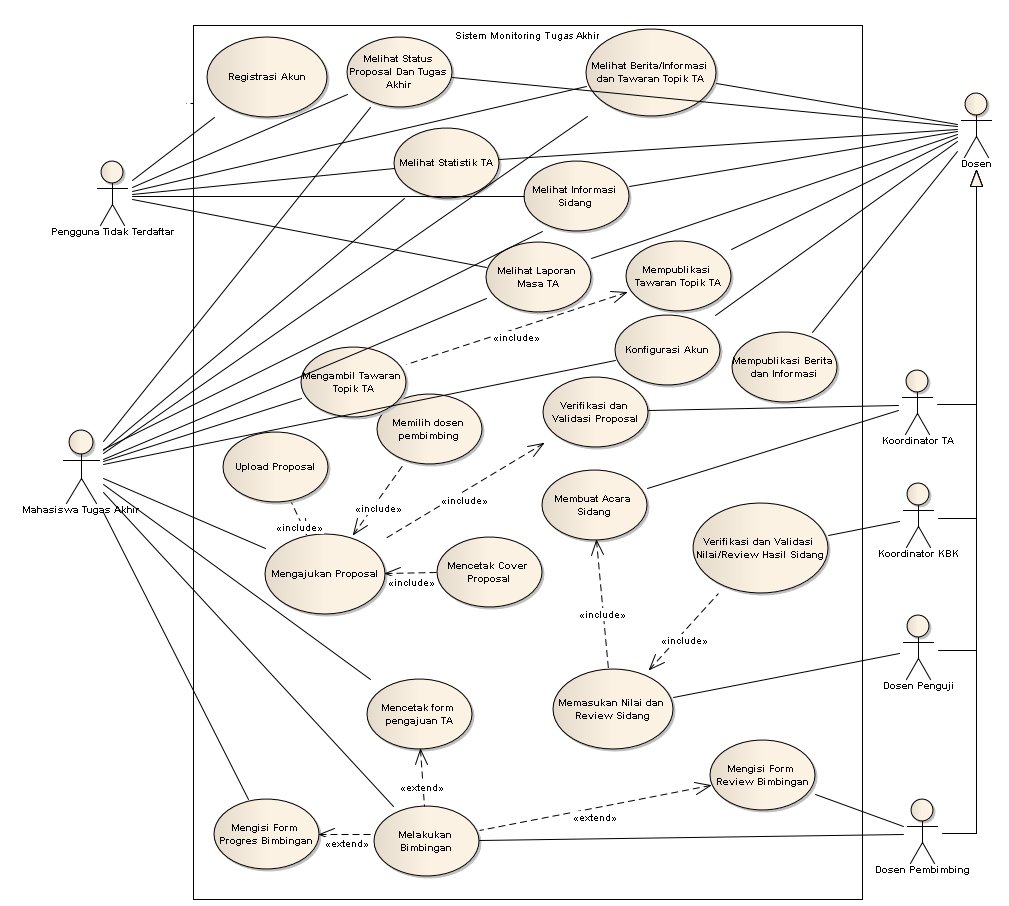
Pada tugas akhir ini, penulis membuat rekayasa perangkat lunak aplikasi sistem monitoring tugas akhir yang menerapkan *incremental rendering view* berbasis teknologi ajax. Perangkat lunak pada sisi klien dibangun dengan menggunakan *Google Web Toolkit* (GWT) yang menggunakan pendekatan *Model View Presenter* (MVP). Perangkat lunak pada sisi server akan dibangun dengan menggunakan framework Ruby on Rails yang menggunakan pendekatan *Model View Controller* (MVC) dengan model data berorientasi obyek. Dengan menggunakan metode MVC, maka aplikasi akan lebih mudah untuk dirawat dan dikembangkan.

Aplikasi Sistem Monitoring Tugas Akhir di Jurusan Teknik Informatika ITS melingkupi semua kegiatan-kegiatan mahasiswa dalam hal pengajuan proposal tugas akhir hingga sidang tugas akhir. Prosedur atau alur pengerjaan tugas akhir tersebut secara garis besar digambarkan pada gambar 3.



**Gambar 3.** Alur pengerjaan tugas akhir mahasiswa*.*

Berdasarkan alur kerja pada gambar diatas, maka deskripsi fungsionalitas dari Aplikasi Sistem Monitoring Tugas Akhir bisa ditunjukan pada gambar sebagai berikut.



**Gambar 4.** Use Case Fungsionalitas Sistem Monitoring Tugas Akhir*.*

Sesuai dengan gambar use case diatas dalam sistem monitoring tugas akhir terdapat beberapa pengguna atau aktor yang berhubungan secara langsung ke dalam sistem. Aksi-aksi yang dilakukan oleh aktor dalam sistem ini meliputi:

1. Melihat Proposal Dan Status Proposal Tugas Akhir

Semua aktor pada sistem bisa melihat laporan atau memonitor status proses pelaksanaan tugas akhir mahasiswa mulai dari pengajuan proposal hingga tugas akhir berakhir.

1. Melihat Informasi Sidang

Semua aktor pada sistem ini bisa melihat Informasi mengenai sidang yang lebih jelasnya mengenai sidang tersebut dijelaskan pada fungsionalitas ke-14.

1. Melihat Berita/Informasi dan Tawaran TA

Semua aktor pada sistem ini dapat melihat Berita/Informasi dan Tawaran TA. Pada Tawaran Topik TA nanti ada fungsi untuk mengambil topik bagi mahasiswa tugas akhir (fungsionalitas ke-7).

1. Registrasi Akun

Registrasi akun dilakukan dengan mengisi form NRP/NIP dan email ITS untuk memastikan pengguna dari ITS. Setelah itu akan diarahkan ke konfigurasi akun (fungsionalitas ke-5).

1. Konfigurasi Akun

Setiap pengguna terdaftar dapat mengatur masing-masing akun yang dimiliki.

1. Mempublikasi Tawaran Topik TA

Setiap dosen yang terdaftar dalam sistem dapat mepublikasi tawaran topik TA pada form yang telah disediakan. Dan akan disediakan laporan mengenai tawaran TA yang dipublikasikan tersebut. Dosen yang mempublikasi dapat mengubah status tawarannya dari *open* (buka) menjadi *close* (tutup), dibekukan (*frozen*), atau batal (*cancel*).

1. Mengambil Tawaran Topik TA

Setiap mahasiswa TA dapat mengambil (*bidding*) dari tawaran topik yang masih bestatus *open.*

1. Mempublikasi Berita dan Informasi

Setiap dosen dapat memberikan pengumuman yang berupa Berita dan Informasi untuk dibaca semua pengguna.

1. Mengajukan Proposal

Mahasiswa tugas akhir hanya bisa mengajukan satu proposal ketika masih ada proposal yang sedang diproses.

1. Memilih Dosen Pembimbing

Calon dosen pembimbing yang dipilih oleh mahasiswa memiliki beberapa ketentuan sebagai berikut:

* Jumlah Dosen Pembimbing TA paling sedikit 1 orang dan paling banyak 2 orang
* Pembimbing 1 adalah Dosen Aktif (tidak berstatus **honorer** atau **TB**)
* Pembimbing 1 harus memiliki **KBK** sesuaidengan Topik TA.
* Pembimbing 2 adalah Dosen yang berkompeten dengan topik TA (boleh dari luar jurusan)

1. Upload Proposal

Susunan proposal yang diupload sesuai dengan pedoman penyusunan Proposal TA.

1. Mencetak Cover Proposal

Cover proposal bisa di download oleh mahasiswa tugas akhir setelah form pengajuan proposal telah dilengkapi dan mengaupload proposal.

1. Verifikasi dan Validasi Proposal

Setiap proposal yang dikumpulkan oleh mahasiswa tugas akhir, maka Koordinator TA akan melakukan verifikasi dan validasi kelengkapan persyaratan sebelum diajukan ke sidang.

1. Membuat Acara Sidang

Koordiantor TA dapat membuat acara sidang yang dilengkapi dengan informasi waktu dan tempat sidang. Informasi sidang akan dikirim ke semua pengguna terdaftar. Pada sidang TA dosen penguji untuk setiap mahasiswa dimasukkan oleh Koordinator TA. Dan dosen penguji pada sidang proposal diambil untuk semua dosen yang hadir pada acara sidang.

1. Memasukkan Nilai dan Review Tugas Akhir

Setiap dosen penguji yang hadir dalam sidang dapat memasukkan nilai dan review dari topik tugas akhir persidangan.

1. Verifikasi dan Validasi Nilai/Review Hasil Sidang

Nilai dan review dari dosen penguji akan diverifikasi dan divalidasi oleh Koordinator KBK sebelum diumumkan hasilnya.

1. Mencetak Form Pengajuan Tugas Akhir

Mahasiswa tugas akhir harus mencetak form pengajuan tugas akhir sebelum diajukan ke sidang.

1. Melakukan Bimbingan

Bimbingan antara mahasiswa tugas akhir dan dosen pembimbing dilakukan minimal 8 kali dengan jangka waktu minimal pengerjaan tugas akhir 3 bulan.

1. Mengisi Form Progres Bimbingan

Sebelum melakukan bimbingan mahasiswa tugas akhir mengisi form progres yang berisi pertanyaan, permasalahan, atau progress dari pengerjaan tugas akhirnya.

1. Melihat Statistik TA

Melihat statistik data Tugas Akhir mahasiswa untuk tiap dosen dan KBK serta ditampilkan pula rata-ratanya.

1. Melihat Laporan Masa TA

Masa pengerjaan TA antara diterimanya proposal hingga sidang TA paling sedikit adalah 3 bulan. Mahasiswa diwajibkan melakukan bimbingan TA paling sedikit 8 kali selama pengerjaan TA (dihitung setelah proposal diterima).

1. Mengisi Form Review Bimbingan

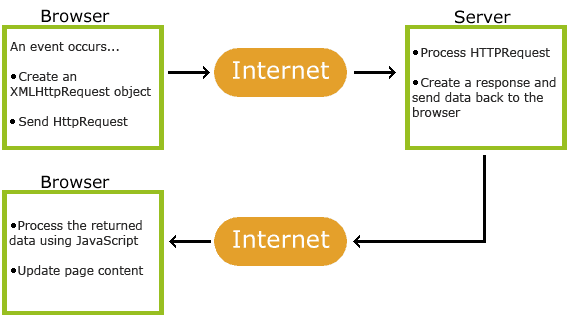
Setelah melakukan bimbingan yang telah direncanakan pada form progres bimbingan mahasiswa, maka dosen pembimbing memberika review mengenai proses bimbingan tersebut.

1. **TINJAUAN PUSTAKA**

Proses pembangunan aplikasi di dalam tugas akhir ini menggunakan beberapa istilah teknologi seperti dibawah ini:

* + - * 1. Ajax

Asynchronous JavaScript and XML (AJAX) merupakan teknik untuk membuat halaman web yang dinamis dan cepat. Ajax memungkinkan halaman diperbaruhi secara *asynchronous* dengan pertukaran data yang jumlahnya kecil dengan server melalui belakang layar. Ini berarti bahwa mungkin untuk memperbaruhi bagian dari suatu web, tanpa memuat ulang seluruh halaman. Pada halaman web tradisional (tanpa menggunakan ajax) harus memuat ulang seluruh halaman yang dikehendaki jika kontennya harus berubah.



**Gambar 5**. Cara kerja AJAX *(diambil dari httphttp://www.w3schools.com/Ajax/ajax\_intro.asp)*

Teknologi ajax dibangun berdasarkan standart internet dan menggunakan kombinasi dari beberapa teknologi:

* + - 1. HTML atau XHTML dan CSS sebagai presentasi.
      2. Document Object Model (DOM) untuk layar dinamis dari dan interaksi dengan data.
      3. XML atau JSON untuk pertukaran data, dan XSLT untuk manipulasinya.
      4. Obyek XMLHttpRequest untuk komunikasi *asynchronous*.
      5. JavaScript untuk mengontrol semua teknologi di atas secara bersama.

Ajax merupakan kumpulan dari teknologi *cross-paltform* yang memungkinkan pengembang untuk membuat halaman web yang berperilaku lebih interaktif layaknya aplikasi desktop. Dengan menggunakan kombinasi teknologi yang telah disebutkan diatas, ajax memungkinkan memperbaruhi halaman seksional dalam menanggapi masukan dari pengguna serta mengurangi proses transfer server (dan waktu tunggu hasilnya) seminimum mungkin. Dengan kode yang sesuai, halaman ajax mengganti halaman lama secara parsial sesuai dengan aksi yang dilakukan pengguna, sehingga meningkatkan waktu respon, interaktivitas, dan *usability*.

1. Google Web Toolkit

Google Web Toolkit (GWT) merupakan perangkat pengembangan untuk membangun dan mengoptimasi aplikasi berbasis browser yang kompleks. GWT mengijinkan untuk mengembangkan dan mendebug aplikasi ajax dalam bahasa java dengan menggunakan perangkat pengembangan Java seperti Eclipse atau NetBeans. GWT menyediakan kompiler Java ke JavaScript dan browser spesial yang membantu untuk mendebug Aplikasi GWT yang dibuat.

Aplikasi GWT yang sedang berjalan dibagi menjadi dua mode, yaitu:

* + - * 1. Mode Hosted

Pada mode ini aplikasi yang dibangun masih berjalan sebagai *bytecode* Java dengan *Java Virtual Machine* (JVM). Sebagian besar waktu dari pengembangan berada pada mode ini, karena aplikasi masih berjalan di JVM. Namun, pada mode ini dimudahkan dengan fasilitas debug dari Java.

* + - * 1. Mode Web

Pada mode ini aplikasi yang dibangun sudah berbentuk JavaScript dan HTML yang dikompile dari kode sumber Java yang asli dengan kompiler Java ke JavaScript. Dan JavaScript dan HTML ini yang nantinya akan di-*deploy* ke web server sehingga *end-user* akan hanya melihat hasil dari mode ini saja.

JRE Emulation Library

(java.lang and java.util)

GWT Web UI class library

**Class Library**

GWT Hosted

Web Browser

GWT Java-to-JavaScript Compiler

**Development Tools**

**Gambar 6.** Arsitektur GWT

Sesuai dalam gambar di atas Arsitektur GWT terdiri dari beberapa komponen utama, yaitu:

1. GWT Java-to-JavaScript Compiler

Pada komponen ini terjadi translasi kode dari bahasa pemrogaman Java ke bahasa pemrogaman JavaScript.

1. GWT Hosted Web Browser

Pada komponen ini mengijinkan untuk menjalankan dan mengeksekusi aplikasi GWT dalam mode hosted, yang mana kode dijalankan sebagai aplikasi Java dalam Java Virtual Machine tanpa mengkompile ke JavaScript.

1. JRE Emulation library

Pada komponen ini berisi implementasi dari sebagian besar kelas yang digunakan dalam library kelas standart.

1. GWT Web UI class library

Pada komponen ini berisi implementasi dari kelas untuk membangun UI dari aplikasi web yang akan dibuat. GWT memungkinkan untuk memanipulasi DOM browser secara langsung menggunakan interface DOM. Ini jauh lebih mudah menggunakan kelas Java dari hirarki widget.

1. JSON-RPC

JSON-RPC merupakan sebuah protocol RPC yang dikodekan dalam format JSON. Ini merupakan protokol yang sangat sederhana dan mirip dengan XML-RPC, hanya mendefinisikan beberapa tipe data dan perintah. JSON-RPC digunakan untuk mekanisme interaksi klien dengan server. JSON-RPC digunakan untuk mengambil data dari server.

Protokol ini memudahkan bagi klien dan server untuk melewati obyek Java bolak-balik melalui HTTP. RPC ini memberikan kesempatan untuk memindahkan semua logika UI ke klien dan meninggalkan logika bisnis di server. Sehingga dengan protokol ini bisa sangat meningkatkan kinerja, mengurangi penggunaan *bandwitdh*, dan mengurangi beban web server.

1. JavaScript Native Interface

Seringkali pada waktu pembangunan aplikasi akan diperlukannya integrasi dengan JavaScript yang dibuat secara *handwritten* atau dengan library JavaScript lainnya. Kadang-kadang diperlukan untuk mengakses fungsi *low-level* dari browser yang tidak ada dalam kelas API GWT. Fitur JavaScript Native Interface (JSNI) pada GWT dapat mengintegrasikan JavaScript secara langsung ke kode sumber Java.

Kompiler GWT menterjemahkan kode sumber Java ke JavaScript. Metode penulisan JSNI adalah sebuah teknik yang *powerful*, karena penulisan method pada JSNI sangat rumit sekali. Namun, kode dari JSNI ini berpotensi kurang portabel di browser, memungkinkan terjadinya *leak memory,* kurang didukung oleh perangkat Java, dan lebih sulit bagi kompiler untuk mengoptimasi.

1. Ruby on Rails

*Ruby on Rails* adalah framework aplikasi web *open source* yang menggunakan bahasa pemrogaman Ruby*.* Ruby on Rails merupakan framework web yang lengkap (*Fullstack*), *Open Source,* dan *Cross-Platform.* Framework web ini menyediakan layar abstraksi database yang sangat baik yang disebut *ActiveRecord*, yang mampu bekerja dengan semua sistem database populer.

Ruby on Rails menggunakan pola arsitektur *Model-View-Controller* (MVC) untuk mengatur pemrogaman aplikasinya. Ruby on Rails mengimplementasikan MVC dengan menyediakan beberapa komponen sebagai bagian dari framework, yaitu:

* *Active Record*

Ini merupakan model dari Ruby on Rails. Komponen model menyimpan data dan menyediakan fungsionalitas untuk bekerja dengan data.

* *Action View*

Komponen *View* meliputi presentasi logik dari data yang ada pada komponen *Model.* *Action View* merupakan komponen *View* pada Ruby on Rails.

* *Active Controller*

Pada komponen ini digunakan untuk menyusun aliran logika. Dalam suatu komponen web, *controller* bertugas mengatur dan menyusun *flow* dari suatu logika aplikasi. *Controller* duduk sebagai batas dari suatu aplikasi dan menangkap semua *request.* Berdasarkan *request, controller* memperbaruhi obyek *Model* yang diinginkan dan memanggil *View Logic* untuk menampilkan data yang telah diperbaruhi. Dalam Ruby on Rails, *Action Controller* menyediakan fungsionalitas-fungsionalitas dari *controller.*

Model

Browser

Controller

Controller

mengirim request

renders

mengirim balik

Mengumpulkan dan update

**Gambar 7.** Arsitektur MVC pada Ruby On Rails

1. **METODOLOGI**

Tahapan yang akan dilakukan dalam tugas akhir ini diantaranya sebagai berikut:

1. **Studi Literatur**

Pada tahap ini penulis mencari tahu mengenai metode-metode apa saja yang bisa digunakan dalam pengembangan web berbasis ajax. Kemudian penulis membandingkan kekurangan dan kelebihan masing-masing metode. Sehingga sampai pada kesimpulan penulis bahwa metode yang paling relevan untuk digunakan adalah metode yang sesuai dengan pengembangan web berbasis ajax pada framework Google Web Toolkit.

Pada tahap ini penulis juga mencari tahu mengenai framework aplikasi web pada Ruby on Rails beserta metode-metode pengembangannya hingga proses *deploy* aplikasi yang dibuat pada server. Selain itu, juga perlu dipelajari metode mengenai pengembangan aplikasi Google Web Toolkit pada Ruby on Rails dengan menggunakan IDE eclipse.

1. **Analisa dan Perancangan Sistem**

Pada tahap ini dilakukan analisa mengenai sistem monitoring tugas akhir dengan menggunakan metode *reverse engineering* untuk mendapatkan informasi mengenai identifikasi sistem, komponen sistem, utilitas serta fungsi-fungsi yang ada pada sistem. Setelah dilakukan analisa tersebut maka diharapkan dapat ditemukan identifikasi atau *recovery* dari kebutuhan program dan atau spesifikasi desain dari aplikasi sehingga dapat membantu dalam memahami dan memodifikasi aplikasi. Pada tahap ini juga dilakukan proses dokumentasi mengenai hasil analisa serta desain dari aplikasi yang akan dibuat.

1. **Implementasi**

Pada tahap ini mulai dibangun aplikasi perangkat lunak sistem monitoring tugas akhir. Pembangunan tersebut dimulai dengan pembentukan antar muka berdasarkan desain pada waktu perancangan dengan menggunakan Google Web Toolkit. Kemudian pada sisi server dibangun mulai dari sistem dasar yang dibutuhkan hingga seluruh fungsional sistem.

1. **Pengujian dan Evaluasi**

Pada tahap ini akan dilakukan pengujian terhadap perangkat lunak menggunakan data atau skenario yang telah dipersiapkan sebelumnya. Uji coba dan evaluasi perangkat dilakukan untuk mencari masalah yang mungkin timbul, mengevaluasi jalannya program, dan mengadakan perbaikan jika ada kekurangan.

1. **Penyusunan Buku Tugas Akhir**

Pada tahap ini melakukan pendokumentasian dan laporan dari seluruh konsep, dasar teori, implementasi, proses yang telah dilakukan, dan hasil-hasil yang telah didapatkan selama pengerjaan tugas akhir. Buku tugas akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan tugas akhir ini dan diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut.

Secara garis besar, buku tugas akhir nantinya terdiri atas beberapa bagian yaitu :

1. Pendahuluan
   1. Latar Belakang
   2. Permasalahan
   3. Batasan Tugas Akhir
   4. Tujuan
   5. Metodologi
   6. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Uji Coba dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka
7. **JADWAL KEGIATAN TUGAS AKHIR**

Tugas akhir ini diharapkan bisa dikerjakan menurut jadwal sebagai berikut:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No. | Kegiatan | 2011 | | | | | |
| Mar | Apr | Mei | Jun | Jul | Agus |
| 1. | Studi Literatur |  |  |  |  |  |  |
| 2. | Analisa dan Perancangan Sistem |  |  |  |  |  |  |
| 3. | Implementasi |  |  |  |  |  |  |
| 4. | Pengujian dan Evaluasi |  |  |  |  |  |  |
| 5. | Penyusunan Buku Tugas Akhir |  |  |  |  |  |  |

1. **DAFTAR PUSTAKA**
2. Yupeng Fu, dkk. 2010. “Ajax-based report pages as incrementally rendered views”. *SIGMOD ‘10*. New York.
3. Daniel Guermeur & Amy Unruh. 2010. “*Google Web Engine Java and GWT Application Development*”. Packt Publishing*,* Birmingham.
4. Dewsbury, Ryan. 2008. “*Google Web Toolkit Applications*”. Prentice Hall, Boston.
5. Web Optimation, LLC. 2010. *Ajax Optimation Chapter 8.* Online. (http://www.websiteoptimization.com/secrets/ajax/. Diakses 11 April 2011)
6. Edi Sugiarto. 2010. *Reverse Engineering.* Online. (http://edisugiarto.blogspot.com/2010/07/reverse-engineering.html. Diakses 10 April 2011)
7. W3schools. 2011. *AJAX Tutorial.* Online. (http://www.w3schools.com/Ajax/default.asp. Diakses 11 April 2011)
8. Sang Shin. 2011. *Google Web Toolkit (GWT).* PDF. (http://*w*ww.javapassion.com/ajax/GWT.pdf. Diakses 11 April 2011)
9. JSON-RPC.org. 2011. *JSON-RPC.* Online. ([http://json-rpc.org/wiki/specification. Diakses 11 April 2011](http://json-rpc.org/wiki/specification.%20Diakses%2011%20April%202011))
10. Cloves Carneiro Jr., & Rida Al Barazi. 2010. “*Beginning Rails 3*”. Apress, New York.

**LEMBAR PENGESAHAN**

###### **Surabaya, 11 Mei 2011**

Menyetujui,

# Dosen pembimbing I

**Dwi Sunaryo, S.Kom., M.Kom.**

**NIP.  197205281997021001**

# Dosen pembimbing II

**Arya Yudhi Wijaya, S.Kom., M.Kom.**

**NIP. 051100119**