

**P2I**

**PROTOTIPE SISTEM MANAJEMEN TUGAS AKHIR BERBASIS WEB  
MENGUNAKAN TEKNOLOGI *JAVA SERVER FACES* (JSF)**



**NASKAH PUBLIKASI**

Oleh:

Gita Indah Marthasari, S.T

Dibiayai oleh Anggaran Dana Pembinaan Pendidikan (DPP) Universitas Muhammadiyah Malang  
Berdasarkan SK Pembantu Rektor I Nomor

LEMBAGA PENELITIAN  
UNIVERSITAS MUHAMMADIYAH MALANG  
Tahun 2009

**HALAMAN PENGESAHAN  
NASKAH PUBLIKASI  
PENELITIAN PENGEMBANGAN IPTEKS**

1. Judul Penelitian : Pembuatan Prototipe Sistem Manajemen Tugas Akhir Berbasis Web Menggunakan Teknologi *Java Server Faces* (JSF)
2. Bidang ilmu penelitian : Rekayasa
3. Ketua Peneliti
- a. Nama Lengkap : Gita Indah Marthasari, S.T
  - b. Jenis Kelamin : P
  - c. NIP : 108.0611.0442
  - d. Pangkat/Golongan : Tenaga Pengajar/IIIA
  - e. Jabatan : -
  - f. Fakultas/Jurusan : Teknik / Teknik Informatika
4. Jumlah Tim Peneliti : 1 orang
5. Lokasi Penelitian : Laboratorium Teknik Informatika UMM
6. Bila penelitian ini merupakan kerjasama kelembagaan
- a. Nama Instansi : -
  - b. Alamat : -
7. Waktu penelitian : 1 Tahun
8. Biaya : Rp 4.000.000 (Empat juta rupiah)

Mengetahui,  
Dekan Fakultas Teknik,

Malang, 18 Juni 2008  
Ketua Peneliti,

Ir. Sunarto, M.T  
NIP. 131885458

Ketua Lembaga Penelitian  
Universitas Muhammadiyah Malang,

Gita Indah Marthasari, S.T  
NIP. 10806110442

Dr. Ir. Maftuchah, M.P  
NIP. 132040895

Pembuatan Sistem Manajemen Tugas Akhir Berbasis Web  
Menggunakan Teknologi *Java Server Faces* (JSF)  
Gita Indah Marthasari  
Universitas Muhammadiyah Malang

## **ABSTRAK**

Tugas akhir adalah tugas yang harus dikerjakan mahasiswa untuk mendapatkan gelar sarjana. Untuk dapat mengambil tugas akhir, seorang mahasiswa harus telah menempuh sedikitnya 110 SKS. Saat ini, seluruh hal yang berkaitan dengan tugas akhir dikelola secara manual. Mulai dari pengajuan proposal, proses pembimbingan, hingga pelaporan. Oleh karena itu, melalui pembangunan sistem pengelolaan tugas akhir berbasis teknologi informasi diharapkan pengelolaan tugas akhir akan lebih efisien. Penelitian ini bertujuan untuk menerapkan Business Process Automation pada pengelolaan tugas akhir sehingga akan terbangun prototipe sistem pengelolaan tugas akhir. Fasilitas yang ada pada prototipe ini hanyalah sebagian dari keseluruhan fasilitas yang dirancang. Melalui penelitian yang berkelanjutan, diharapkan sistem ini dapat selesai secara utuh.

Kata-kata kunci : tugas akhir, web, informatika, jsf, otomasi, java server faces

## **SUMMARY**

Undergraduate thesis is a task that must be done by an university students to get a degree. To be able to take the thesis, a student must have been through at least 110 SKS. At this time, all things relating to the management of thesis is managed manually. Therefore, through the development of automation of undergradute thesis management system, the management is expected to be more efficient. This research aims to implement the Business Process Automation in the management of undergraduate thesis. Facilities that exist in the prototype this is just part of the overall facilities are designed. Through continuous research, it is expected that this system can be completely done.

Keywords : undergraduate thesis, web, informatic, jsf, automation, java server faces

## PENDAHULUAN

Saat ini website telah menjadi salah satu alat utama pertukaran informasi. Hal ini disebabkan antara lain karena melalui website penyebaran informasi dapat berlangsung secara efektif dan efisien. Masyarakat yang dulunya awam dengan penggunaan website kini telah semakin pandai menggunakannya dan didukung oleh fasilitas koneksi internet yang semakin mudah diperoleh. Adanya beragam fitur yang disediakan oleh sebuah website juga turut andil dalam peningkatan popularitas website di mata masyarakat.

Teknologi pengembangan website juga sangat beragam. HTML, PHP, Java Script, ASP, JSP, CGI, dan XML adalah beberapa contoh teknologi tersebut. Masing-masing memiliki instruksi-instruksi khusus untuk pembuatan website, baik website statik maupun website dinamik. Website statik adalah website yang hanya menampilkan informasi statik dan tidak berubah-ubah, sedangkan website dinamik adalah website yang tampilannya tergantung kepada masukan user.

JSP atau Java Server Pages adalah teknologi pengembangan website yang bersifat *Java-based*. JSP adalah teknologi pengembangan website berbasis Java setelah Servlet. Namun, penggunaan JSP saat ini sudah mulai ditinggalkan karena muncul teknologi pengembangan yang lebih baru. Teknologi baru ini mengakomodasi prinsip MVC (*Model-View-Controller*) yang memecah aplikasi menjadi bagian Model, View, dan Control [1]. Dengan menerapkan paradigma ini, pengembangan aplikasi dapat menjadi lebih efektif karena perubahan pada salah satu komponen tidak akan berpengaruh terhadap komponen lainnya. Salah satu teknologi yang menggunakan paradigma MVC ini adalah JSF (Java Server Faces).

JSF adalah framework untuk memberikan kemudahan pada proses pengembangan antarmuka pengguna pada aplikasi web berbasis Java. JSF adalah perangkat lunak open source yang dikembangkan oleh perusahaan Sun Microsystems. JSF memudahkan pengembang aplikasi web yang telah terbiasa menggunakan GUI (*Graphical User Interface*) Java.

Tugas akhir adalah tugas yang wajib dikerjakan oleh tiap mahasiswa Jurusan Teknik Informatika. Setiap mahasiswa yang mengambil tugas akhir akan menentukan sendiri topik yang akan dikerjakan dan akan didampingi oleh minimal seorang dosen pembimbing. Data-data yang terkait dengan seluruh proses pengerjaan tugas akhir ini perlu disimpan dan dikelola dengan tepat sehingga informasi-informasi yang terkait dapat diakses dengan mudah di kemudian hari. Sistem pengelolaan tugas akhir ini juga nantinya akan dapat memberikan kemudahan kepada mahasiswa yang memerlukan informasi mengenai tugas akhir seperti topik tugas akhir yang ditawarkan, dosen pembimbing, jadwal pembimbingan, dan lain sebagainya.

## **METODE PENELITIAN**

Pendekatan yang digunakan dalam penelitian ini dikelompokkan menjadi tahapan berikut :

### **1. Studi pustaka dan pengumpulan data**

Hal-hal yang dilakukan pada tahapan ini adalah :

- Mengumpulkan informasi mengenai prosedur pengerjaan tugas akhir di Jurusan Teknik Informatika
- Mengumpulkan literatur yang diperlukan baik dari internet maupun dari buku-buku teks mengenai implementasi JSF untuk pembuatan website.
- Mencari dan mempelajari library-library yang diperlukan pada proses implementasi perangkat lunak.
- Mempelajari teknik implementasi JSF pada perangkat lunak yang akan digunakan pada tahapan implementasi desain

### **2. Melakukan analisa *business process***

Berdasarkan informasi yang diperoleh dari tahap sebelumnya, dilakukan analisa untuk menghasilkan spesifikasi kebutuhan perangkat lunak yang akan dibangun. Spesifikasi yang dicakup adalah spesifikasi fungsional minimal yang berisi kemampuan atau fitur apa saja yang nantinya dimiliki oleh sistem dan spesifikasi tampilan yaitu bagaimana tampilan yang diinginkan pengguna terhadap sistem. Dari hasil analisa nantinya dapat diketahui siapa saja pengguna sistem dan fitur apa saja yang dibutuhkan oleh seluruh pengguna sistem.

### 3. Membuat desain website

#### a. Desain sistem

Perancangan sistem akan disesuaikan dengan spesifikasi fungsional yang dihasilkan pada tahap sebelumnya. Perancangan sistem akan dilakukan dengan teknik *object-oriented* menggunakan UML (*Unified Modelling Language*) untuk menggambarkan fungsionalitas sistem (menggunakan *use-case diagram* dan *activity diagram*) dan basis data (menggunakan *class-diagram*).

#### b. Desain tampilan website

Desain tampilan web disesuaikan dengan spesifikasi tampilan yang dihasilkan dari tahap sebelumnya.

### 4. Mengimplementasikan desain perangkat lunak

Perangkat lunak yang digunakan untuk implementasi desain adalah sebagai berikut :

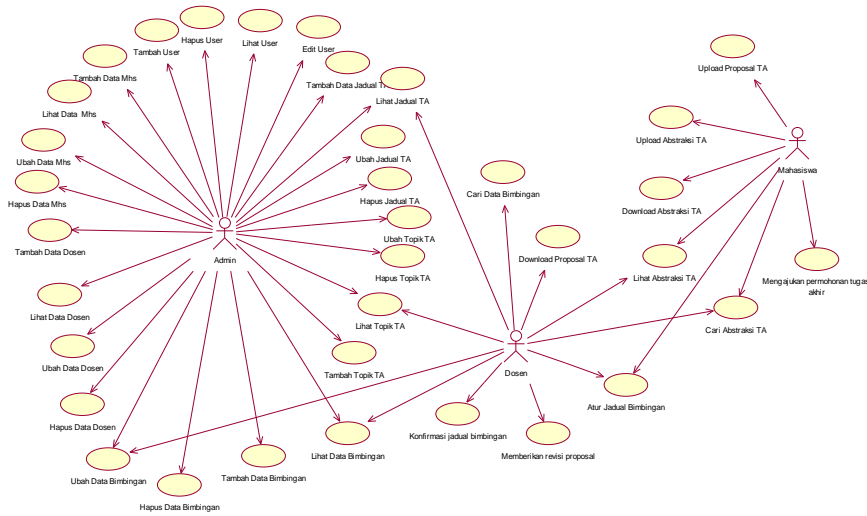
1. JDK 1.6
2. Mojarra
3. Apache Tomcat 6.0
4. MySQL 5.0
5. Eclipse Europa

### 5. Pengujian perangkat lunak

Pengujian perangkat lunak dilakukan untuk memastikan bahwa perangkat lunak telah sesuai dengan kebutuhan yang dispesifikasikan pada tahap analisa.

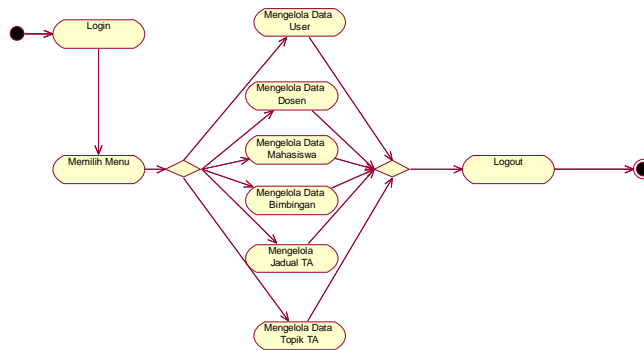
## **HASIL DAN PEMBAHASAN**

Berdasarkan hasil analisa, diperoleh spesifikasi fungsionalitas sistem sebagai berikut :



Gambar 1 Use Case Diagram Sistem yang akan dibangun

Adapun diagram aktivitas dari sistem adalah sebagai berikut :

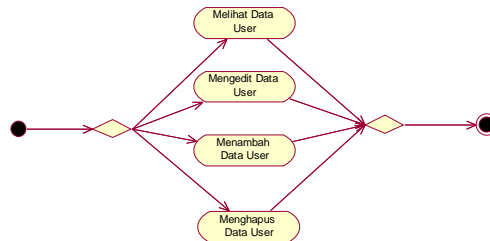


Gambar 2 Activity Diagram sistem yang akan dibangun

Berdasarkan gambar di atas, terdapat beberapa aktivitas yang di dalamnya terdiri atas beberapa aksi, yaitu : mengelola data user, mengelola data dosen, mengelola data mahasiswa, mengelola data bimbingan, mengelola jadwal tugas akhir, dan mengelola topik tugas akhir. Untuk dapat menjalankan aktivitas-aktivitas tersebut, administrator diharuskan melakukan

login. Detail mengenai masing-masing aktivitas dapat digambarkan dalam beberapa gambar diagram aktivitas berikut :

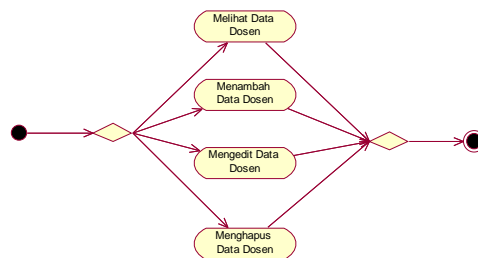
### 1. Aktivitas mengelola data user



Gambar 3 Activity Diagram pengelolaan user

Berdasarkan diagram aktivitas di atas, maka yang dapat dilakukan oleh admin dalam rangka mengelola data user antara lain : melihat, mengedit, menambah, dan menghapus data user.

### 2. Aktivitas mengelola data dosen

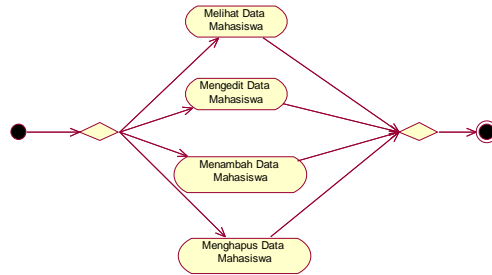


Gambar 4 Activity Diagram Pengelolaan Dosen

Berdasarkan gambar di atas, maka aksi yang dapat dilakukan pada aktivitas mengelola data dosen adalah melihat, menambah, mengedit, dan menghapus data dosen.

### 3. Aktivitas mengelola data mahasiswa

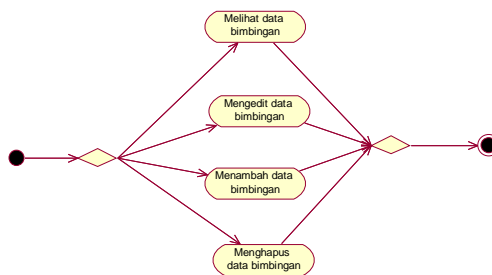




Gambar 5 Activity Diagram Pengelolaan Mahasiswa

Berdasarkan diagram aktivitas di atas, maka proses yang berlangsung pada aktivitas mengelola data mahasiswa adalah melihat, mengedit, menambah, dan menghapus data mahasiswa.

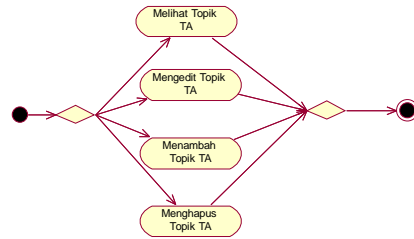
#### 4. Aktivitas mengelola data bimbingan



Gambar 6 Activity Diagram Pengelolaan Data Bimbingan

Berdasarkan gambar di atas, proses yang berlangsung pada aktivitas mengelola data bimbingan adalah melihat, mengedit, menambah, dan menghapus data bimbingan.

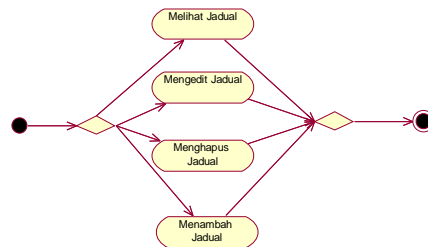
#### 5. Aktivitas mengelola data topik tugas akhir



Gambar 7 Activity Diagram Pengelolaan Topik Tugas Akhir

Berdasarkan gambar di atas, maka aksi yang dapat dilakukan pada aktivitas mengelola topik tugas akhir adalah melihat, menambah, mengedit, dan menghapus data topik tugas akhir.

#### 6. Aktivitas mengelola data jadwal tugas akhir



Gambar 8 Activity Diagram Pengelolaan Jadwal

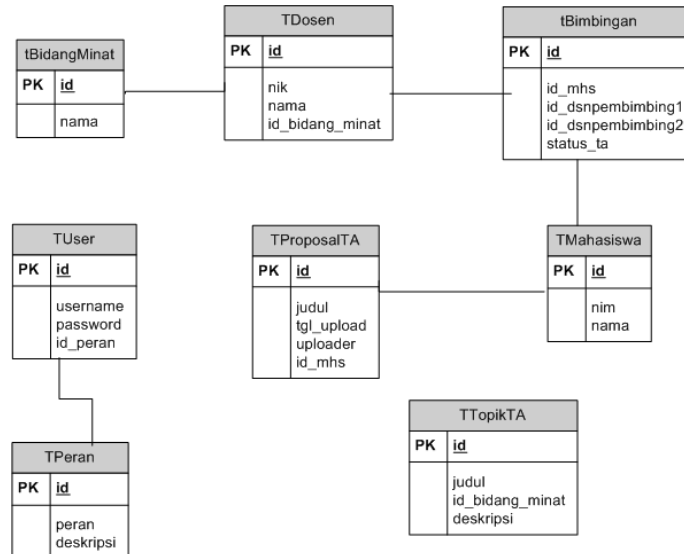
Berdasarkan gambar di atas, maka aksi yang dapat dilakukan pada aktivitas mengelola jadwal tugas akhir adalah melihat, menambah, mengedit, dan menghapus data jadwal tugas akhir.

Berdasarkan hasil analisa di atas, maka data-data yang perlu dikelola oleh sistem meliputi :

- Data dosen
- Data mahasiswa
- Data pengguna sistem (*user*)
- Data bimbingan baik dari sisi dosen maupun mahasiswa

- Data topik tugas akhir
- Jadwal kegiatan yang berkaitan dengan tugas akhir

Rancangan skema basisdata dapat dilihat pada Entity Relationship Diagram berikut :



Gambar 9 Rancangan basis data sistem

Prototipe yang akan dibangun memiliki struktur menu sebagai berikut :

1. Pengelolaan user
  - 1.2 Mengedit data user : mengganti username dan password
  - 1.3 Menghapus data user
  - 1.4 Menambah user baru
2. Pengelolaan dosen
  - 2.1 Mengedit data dosen : mengganti bidang minat/keahlian
  - 2.2 Menghapus data dosen
  - 2.3 Menambah data dosen
3. Pengelolaan mahasiswa
  - 3.1 Mengedit data mahasiswa : mengedit nama

3.2 Menghapus data mahasiswa

3.3 Menambah data dosen

4. Pengelolaan bimbingan

4.1 Mengedit data bimbingan : mengedit status tugas akhir, nama dosen pembimbing 1, nama dosen pembimbing 2

4.2 Menghapus data bimbingan per mahasiswa

5. Pengelolaan topik tugas akhir

5.1 Menambah topik tugas akhir

5.2 Mengedit topik tugas akhir : mengedit judul topik dan bidang minat

5.3 Menghapus topik tugas akhir

6. Pengelolaan jadwal tugas akhir

5.1 Menambah jadwal baru dengan mengupload file yang berisi jadwal kegiatan tugas akhir

5.2 Mengedit informasi terkait masing-masing jadwal

5.3 Menghapus jadwal

Rancangan antarmuka prototipe sistem yang akan dibangun adalah sebagai berikut :



Gambar 10 Rancangan tampilan antarmuka

Pada tahapan implementasi dilakukan pengkodean prototipe berdasarkan hasil dari tahap perancangan. Proses pengkodean dilakukan untuk tiga komponen framework, yaitu komponen model yang berupa kelas-kelas Java, komponen view yang berupa file-file JSP, dan komponen controller yang berupa file-file XML.

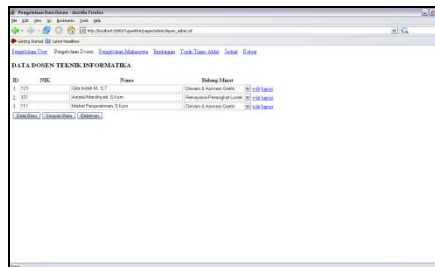
Beberapa di antara halaman pada web adalah sebagai berikut :

### 1. Halaman pengelolaan user



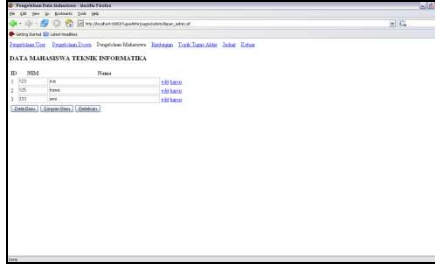
Gambar 11 Tampilan halaman pengelolaan user

### 2. Halaman pengelolaan dosen



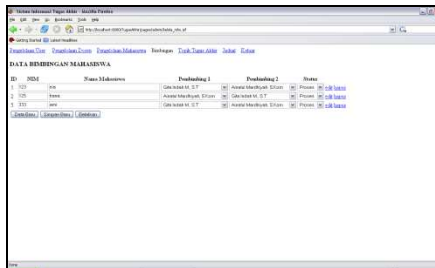
Gambar 12 Tampilan halaman pengelolaan dosen

### 3. Halaman pengelolaan mahasiswa



Gambar 13 Tampilan halaman pengelolaan mahasiswa

#### 4. Halaman pengelolaan bimbingan



Gambar 14 Tampilan halaman pengelolaan bimbingan

#### 5. Halaman pengelolaan topik tugas akhir



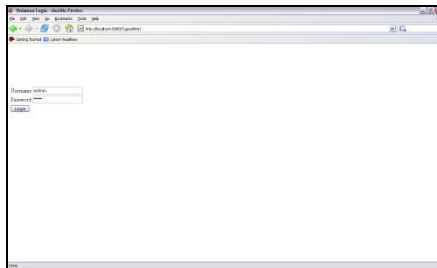
Gambar 15 Tampilan halaman pengelolaan topik tugas akhir

#### 6. Halaman pengelolaan jadwal



Gambar 16 Tampilan halaman pengelolaan jadwal

## 7. Halaman Login



Gambar 17 Halaman Login

## KESIMPULAN DAN SARAN

Kesimpulan yang dapat diambil dari penelitian ini adalah bahwa teknologi *Java Server Faces* (JSF) efektif digunakan untuk membangun sistem yang berskala besar karena dengan teknologi ini pembangunan sistem dapat dengan mudah didekomposisi menjadi modul-modul yang dikembangkan secara terpisah. Modul-modul terpisah ini dapat dikembangkan tanpa ada kesaling-tergantungan antara satu dengan lainnya karena masing-masing modul akan membahas komponen yang berlainan, yaitu : komponen model, *view*, dan *controller*.

Adapun saran bagi penelitian ini adalah :

- prototipe dapat dilengkapi dengan fitur-fitur untuk dosen dan mahasiswa peserta tugas akhir
- tampilan antarmuka yang lebih menarik
- pengelolaan user session

## DAFTAR PUSTAKA

1. JENI-Web Programming-Bab 7-MVC Intro.pdf
2. <http://dhiku.wordpress.com/2008/05/18/road-to-jsf-part-i/>, diunduh pada tanggal 2 Juni 2008.
3. Wikipedia : Java Server Faces. [http://en.wikipedia.org/wiki/JavaServer\\_Faces.htm](http://en.wikipedia.org/wiki/JavaServer_Faces.htm), diunduh pada tanggal 2 Juni 2008.
4. JENI-Web Programming-Bab 9-JSF.pdf
5. Wikipedia : Model-view-controller. <http://en.wikipedia.org/wiki/Model-view-controller>, diunduh pada tanggal 27 Juni 2008
6. <http://www.developersbook.com/jsf/jsf-tutorial/jsf-tutorial.php>, diunduh pada tanggal 27 Juni 2008
7. Balusc. The Balusc Code : Using Datatable <http://balusc.blogspot.com/2006/06/using-datatables.html>, diakses pada tanggal 27 Januari 2009
8. Balusc. The Balusc Code : Communication in JSF. <http://balusc.blogspot.com/2006/06/communication-in-jsf.html>. Diakses pada tanggal 27 Januari 2009.