

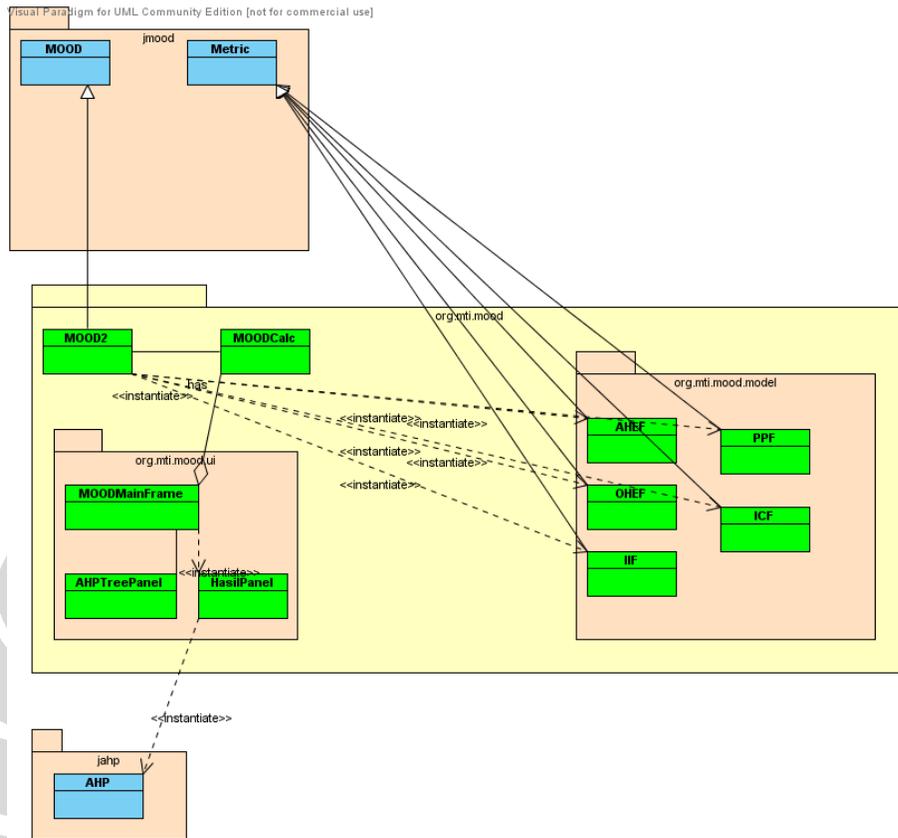
## BAB 3

# METODOLOGI PENELITIAN

Bab ini akan menjelaskan tentang desain alat bantu yang dikembangkan, serta metode percobaan yang akan dilakukan beserta sampel yang digunakan untuk percobaan tersebut.

### 3.1. Desain Alat Bantu

Alat bantu yang dikembangkan akan melanjutkan pengembangan yang telah dilakukan penelitian sebelumnya dengan melakukan *extensions* terhadap *class-class* yang telah ada. Pengembangan dilakukan dengan membagi alat bantu menjadi tiga paket, yaitu `org.mti.mood` sebagai paket utama, `org.mti.mood.model` sebagai paket yang berisikan penyedia data untuk antarmuka maupun untuk menampung metrik-metrik yang baru, serta `org.mti.mood.ui` sebagai paket antarmuka. Selain paket-paket baru, alat bantu yang dikembangkan juga menggunakan *library* yang didapatkan dari hasil penelitian sebelumnya, yaitu *library jmood*<sup>[DES08]</sup>. Gambar 3.1 adalah *class diagram* yang menggambarkan hubungan antar kelas dalam alat bantu yang dikembangkan:



**Gambar 3.1 Class Diagram jMOODCalculator**

Selain pembuatan *class* dan paket baru (latar belakang kuning terang), juga dilakukan perubahan terhadap *class* COF pada paket *jmood*, untuk mengakomodasi perubahan perhitungan yang dilakukan pada metrik MOOD2. *Class* MOOD2 adalah turunan dari *class* MOOD pada paket *jmood*, *class* ini menambahkan fungsi perhitungan untuk metrik-metrik MOOD2. Selain itu juga dilakukan perubahan terhadap pembacaan *input* perangkat lunak yang akan diuji, supaya dapat melakukan pembacaan perangkat lunak yang terbagi menjadi beberapa paket dalam satu kali pembacaan.

### 3.2. Desain Eksperimen

Proses eksperimen dilakukan dengan mengumpulkan sampel perangkat lunak yang akan diukur kualitasnya berdasarkan metrik MOOD2. Program yang akan diukur semua berbasis bahasa pemrograman Java. Setelah sampel dikumpulkan, maka masing-masing perangkat lunak akan digunakan sebagai input untuk alat bantu yang telah dibangun, kemudian hasil perhitungan dari seluruh sampel akan ditampilkan.

Hasil perhitungan yang berupa daftar nilai metrik MOOD2 untuk tiap sampel perangkat lunak akan digunakan sebagai masukan untuk tahap berikutnya yaitu pengukuran dengan AHP. Tiap metrik dalam MOOD2 akan menjadi kriteria pengukuran AHP, sementara nilai masing-masing metrik akan menjadi nilai prioritas lokal. Langkah berikutnya adalah menentukan *pairwise comparison* untuk tiap kriteria pengukuran, dan menghitung nilai prioritas global. Hasil perhitungan nilai akhir untuk tiap sampel perangkat lunak berdasarkan hirarki penilaian yang telah ditentukan akan ditampilkan. Khusus untuk nilai metrik CCF, eksperimen juga akan mencoba mengukur nilai  $(1 - CCF)$ , karena untuk nilai CCF yang semakin kecil menunjukkan kualitas desain yang lebih baik.

Tabel 3.1 menunjukkan spesifikasi perangkat yang digunakan untuk melakukan eksperimen.

Tabel 3.1 Spesifikasi Perangkat

Komponen	Perangkat Keras

### 3.2.1 Sampel Program

Sampel program yang akan diukur dalam eksperimen adalah program-program ERP berbasis Java, empat di antaranya adalah program yang digunakan dalam penelitian Des<sup>[DES08]</sup> sebagai pembanding, ditambah dengan tiga buah program ERP lain, serta paket `java.util`, `java.io`, `java.math`, `java.text`, `java.lang` dan `java.net` dari *java 2 standard edition library* Sun.

- Paket `java.util`  
Paket `java.util` adalah bagian dari Sun Java Platform Standard Edition, yang berisi *collections framework*, *legacy collection classes*, model untuk penanggalan dan waktu, *internationalization*, serta *utility classes* seperti *class* pemecah *string*, manipulasi bit dan *random number generator*.
- Paket `java.io`  
Paket `java.io` berisi koleksi *class* yang berurusan dengan masalah *input-output* sistem melalui *data streams*, *serialization* dan *file system*.
- Paket `java.math`  
Paket `java.math` berisi koleksi *class* yang membantu pengembang perangkat lunak yang membutuhkan perhitungan matematik dengan presisi yang lebih tinggi untuk bilangan *integer* dan *decimal*.
- Paket `java.text`  
Paket `java.text` menyediakan koleksi *class* untuk menangani teks, tanggal, angka dan pesan yang independen dari bahasa *natural*.
- Paket `java.lang`  
Paket `java.lang` menyediakan koleksi *class* dasar untuk bahasa pemrograman Java, termasuk *class* `Object` yang merupakan *parent* dari seluruh *class* yang ada, serta *class-class* tipe dasar seperti `String`, `Integer`, `Long` dan lain sebagainya.
- Paket `java.net`

Paket *java.net* berisi koleksi *class* untuk membantu implementasi perangkat lunak yang menggunakan jaringan, termasuk untuk aplikasi berbasis *web*. Terdapat dua bagian paket ini, yaitu *Low Level API* dan *High Level API*.

- **Adempiere**

Adempiere adalah sebuah perangkat lunak berbasis *open source* yang mulai ada sejak September 2006. Perangkat lunak ini adalah *fork* dari perangkat lunak Compiere, dan mencakup fungsi-fungsi *Enterprise Resource Planning (ERP)*, *Supply Chain Management (SCM)*, *Customer Relationship Management (CRM)*, *Financial Performance Analysis*, *Point of Sales* dan *Webstore*. Pengembangan dilakukan secara terdistribusi melalui jaringan internet, dan untuk repositori-nya terdapat di situs SourceForge ([www.sourceforge.net](http://www.sourceforge.net)).

- **Compiere**

Compiere adalah perangkat lunak *open source* untuk fungsi ERP dan CRM yang ditujukan untuk penggunaan oleh usaha kecil dan menengah pada bidang distribusi, retail, jasa dan manufaktur. Selain versi gratis, Compiere juga menawarkan lisensi komersil yang memiliki layanan tambahan seperti *training* dan *support*. Compiere bisa didapatkan melalui perusahaan Compiere Inc. ([www.compiere.com](http://www.compiere.com)).

- **Freedom ERP**

Freedom adalah perangkat lunak *free* untuk manajemen bisnis. Modul-modul perangkat lunak ini ada banyak, mulai dari modul *standard*, *financial*, *production*, *telemarketing*, *pos* dan *service*. Repositori pengembangan aplikasi ini juga bisa didapatkan melalui situs SourceForge.

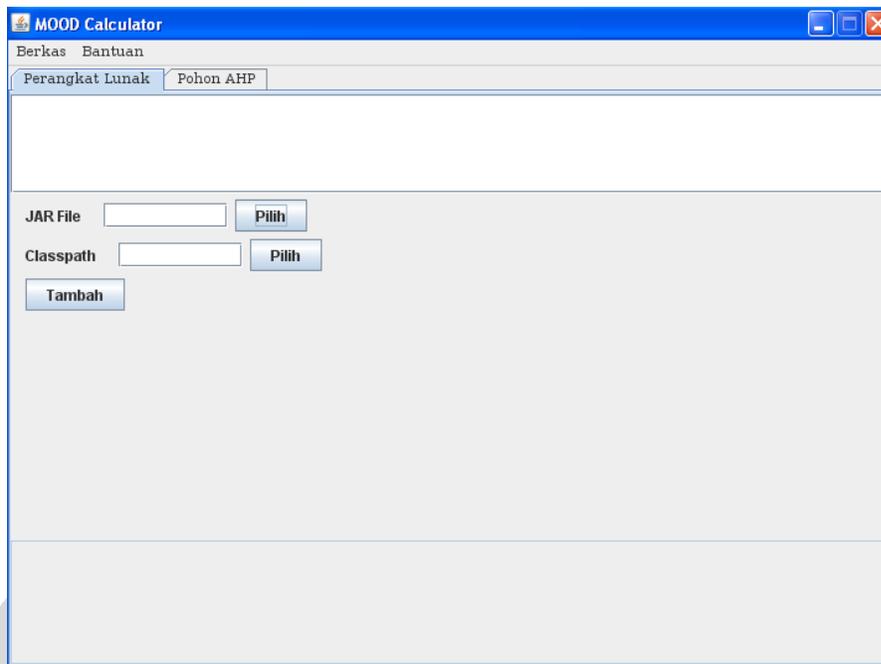
- **jAllInOne**

jAllInOne merupakan perangkat lunak ERP dan CRM yang dikembangkan sebagai RIA (*Rich Internet Application*) dengan protokol HTTP sebagai perantara komunikasi antara *client* dan *server*. Seperti kebanyakan perangkat lunak yang diuji, repositori pengembangan jAllInOne juga tersedia di SourceForge.

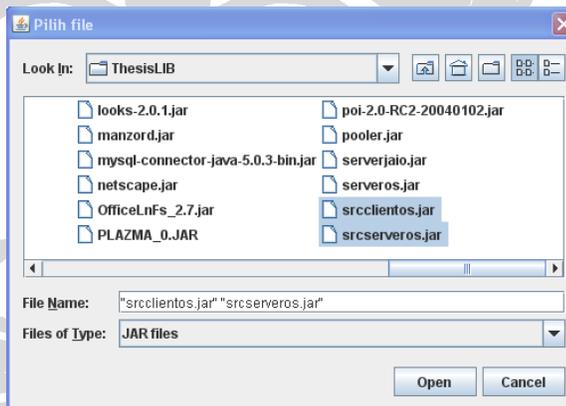
- **MilleniumBSA**  
Millenium *Business Suite Anywhere* (BSA) adalah aplikasi ERP/CRM berbasis web yang memiliki fitur-fitur *resource planning, distribution, inventory, payroll, human resources, purchasing* dan *sales*. Lisensinya adalah GPLv2, dan dapat diunduh dari situs SourceForge.
- **Plazma**  
Plazma menyediakan berbagai modul untuk aplikasi bisnisnya, di antaranya adalah *account management, contacts management, products management, sales management, tasks management, campaigns management* serta *analytical reports*. Repositori perangkat lunak ini tersedia di SourceForge.
- **TNTConcept**  
TNTConcept adalah perangkat lunak yang memungkinkan penggunaanya untuk melakukan manajemen *customer, provider, contacts, business interaction, projects, cash flow*, dan lain sebagainya, secara terintegrasi. Aplikasi ini menggunakan teknologi J2EE terbaru, yaitu JSF, Spring Framework, Hibernate ORM dan Maven build tool. Repositori perangkat lunak ini ada di SourceForge.

### 3.2.2 Alat Bantu Pengukuran Kualitas Desain

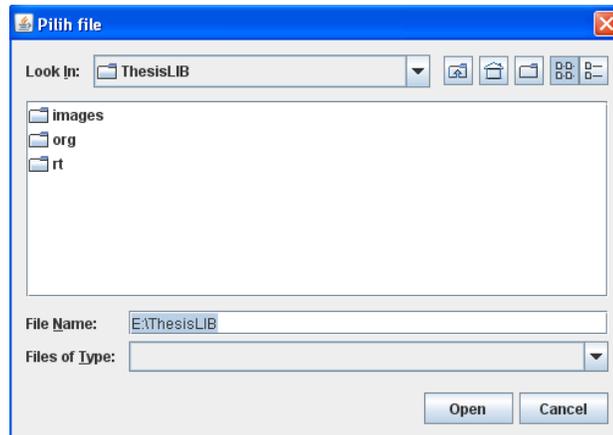
Berikut ditampilkan beberapa gambar yang menunjukkan proses penggunaan alat bantu untuk pengukuran kualitas desain. Gambar 3.2 dan Gambar 3.3 menunjukkan tampilan utama alat bantu dan pemilihan perangkat lunak yang akan diukur, sementara Gambar 3.4 menampilkan pemilihan *classpath* untuk *library* yang dibutuhkan oleh perangkat lunak tersebut:



**Gambar 3.2 Tampilan Utama Alat Bantu**

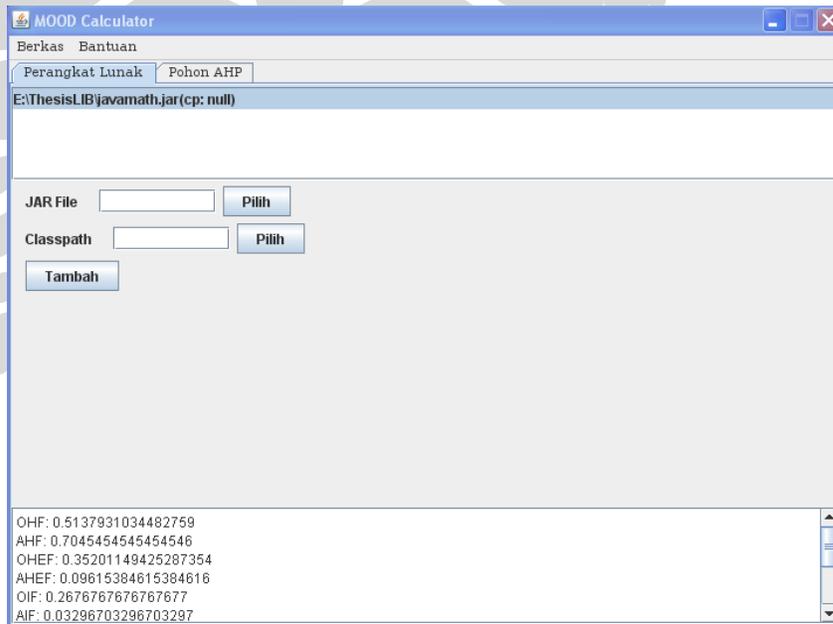


**Gambar 3.3 Pemilihan Perangkat Lunak**



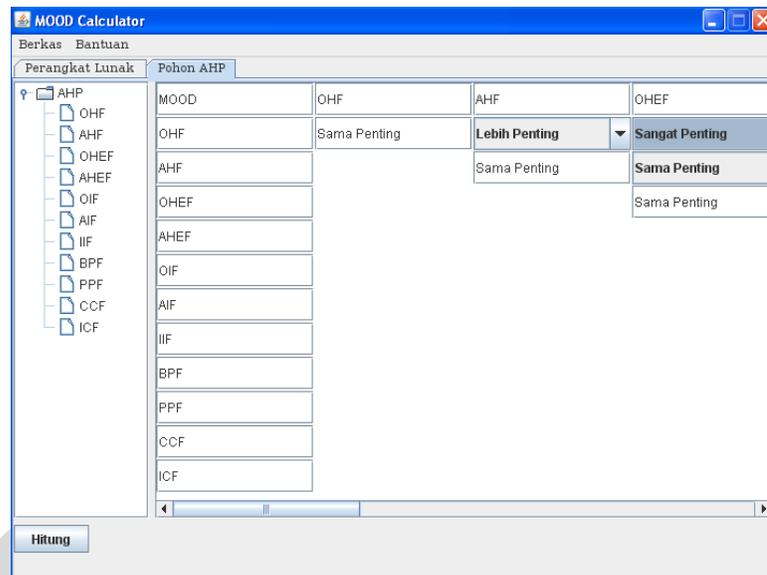
**Gambar 3.4 Pemilihan Classpath**

Setelah memilih perangkat lunak yang akan diuji dan *classpath*-nya, maka alat bantu akan memasukkan perangkat lunak dan *classpath* tersebut ke dalam daftar yang akan diukur. Nilai hasil penghitungan metrik MOOD2 dapat dilihat dengan memilih salah satu perangkat lunak yang ada di dalam daftar, seperti pada Gambar 3.5.



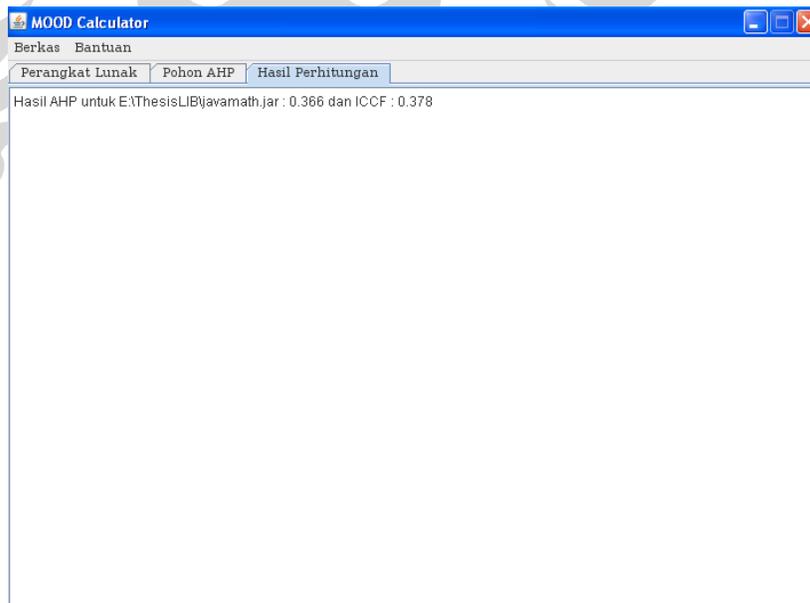
**Gambar 3.5 Hasil Penghitungan Metrik MOOD2**

Langkah selanjutnya adalah melakukan konfigurasi terhadap *pairwise comparison matrix* dengan memilih tab 'Pohon AHP'. Tampilan konfigurasi seperti pada Gambar 3.6:



**Gambar 3.6 Konfigurasi *Pairwise Comparison Matrix***

Setelah menentukan konfigurasi *pairwise comparison matrix* pengguna alat bantu menekan tombol hitung untuk menampilkan hasil perhitungan AHP untuk masing-masing perangkat lunak yang diuji, nilai metrik MOOD2 dan modifikasi (1 – CCF). Urutan penampilan hasil sesuai urutan *input* perangkat lunak, seperti tampak pada Gambar 3.7:



**Gambar 3.7 Hasil Perhitungan AHP**

