

BAB III

METODOLOGI

Pada bab ini dijelaskan langkah-langkah yang akan dilakukan dalam perancangan, implementasi dan pengujian dari aplikasi perangkat lunak yang akan dibuat. Kesimpulan dan saran disertakan sebagai catatan atas aplikasi dan kemungkinan arah pengembangan aplikasi selanjutnya.

3.1 Studi Literatur

Studi literatur menjelaskan dasar teori yang digunakan untuk menunjang penulisan penelitian. Teori pendukung tersebut meliputi:

- a. Sistem Pendukung Keputusan (*Decision Support System (DSS)*)
- b. *Data Mining*
- c. *K-Nearest Neighbor*
- d. Rekayasa Perangkat Lunak:
 - Analisis Kebutuhan (*Requirement Analysis*)
 - Perancangan (*Design*)
 - Implementasi
 - Pengujian (*Testing*)

3.2 Pengumpulan Data

Data yang digunakan pada penelitian ini diperoleh dengan cara mengambil nilai matakuliah dari beberapa mahasiswa angkatan 2007 dan 2008 yang telah menempuh penjurusan. Parameter data yang akan digunakan dalam penelitian ini berupa nilai matakuliah prasyarat dari masing-masing konsentrasi, antara lain:

1. Analisis dan Perancangan Sistem
2. Grafika Komputer
3. Pemrograman Berorientasi Objek
4. Rekayasa Perangkat Lunak
5. Sistem Digital dan Mikroprosesor
6. Pemrograman Lanjut

7. Kecerdasan Buatan
8. Basis Data Lanjut
9. Pengolahan Citra Digital
10. Pengenalan Pola
11. Probabilitas dan Statistika
12. Jaringan Komputer

3.3 Penerapan Algoritma *K-Nearest Neighbor*

Penerapan algoritma *K Nearest Neighbor* dalam penelitian ini akan digunakan sebagai proses perhitungan atas *input* yang telah diberikan oleh *user*. Dari jawaban yang dimasukkan *user*, dilakukan proses perhitungan terhadap data input berdasarkan basis pengetahuan yang ada pada sistem dengan menggunakan metode *K Nearest Neighbor*, yaitu:

- Menentukan *K*, yaitu dengan memilih sejumlah *K* tetangga terdekat.
- Menghitung jarak atribut setiap data tes dengan semua atribut *data training* dengan *Euclidean Distance* [NIN-06:69]:

$$d(x, y) = \sqrt{\sum_{k=1}^n (x_k - y_k)^2}$$

- Dari perhitungan *Euclidean Distance* dilakukan pengurutan data, mulai dari jarak yang paling kecil hingga yang paling besar.
- Data baru dimasukkan ke dalam kategori dengan jarak yang paling dekat berdasarkan mayoritas yang ada pada *K* sebagai hasil keputusan dari sistem.
- Untuk mengukur tingkat kecocokan, maka *user* memasukkan keminatan pada konsentrasi tertentu kemudian dihitung dengan:

$$\% \text{ kecocokan} = (\text{jumlah data yang cocok dari } K / K) * 100$$

3.4 Pengembangan Perangkat Lunak

Tahap pengembangan perangkat lunak dalam pembuatan aplikasi ini menggunakan *classic life style* atau yang lebih dikenal dengan istilah *waterfall*.

3.4.1 Analisis Kebutuhan

Analisis kebutuhan dilakukan dengan menentukan kebutuhan apa saja yang dibutuhkan untuk membangun sistem pendukung keputusan. Metode analisis yang digunakan adalah *Object Oriented Analysis* dengan menggunakan bahasa pemodelan UML (*Unified Modelling Language*). Diagram *use case* digunakan untuk mendepelintikan jalannya sistem sebagai respon dari *request* yang berasal dari luar sistem dan bagaimana *actor* berinteraksi dengan sistem. Skenario yang akan dijalankan dalam sistem ini adalah:

- *User* memasukkan *input* berupa nilai mata kuliah.
- Sistem akan memproses masukan dari *user* dengan perhitungan *K Nearest Neighbor*.
- *User* akan menerima *output* berupa hasil analisis sistem dan prosentase kecocokan terhadap salah satu konsentrasi.

Kebutuhan yang digunakan dalam pembuatan sistem pendukung keputusan ini meliputi:

1. Kebutuhan *Hardware*, meliputi:
 - Komputer PC
2. Kebutuhan *Software*, meliputi:
 - Microsoft Windows XP sebagai sistem operasi
 - My Sql sebagai server *Database Management System*
 - Netbeans 6.9 sebagai *Integrated Development Environtment*
 - JSE (*Java Standard Edition*) 6 sebagai platform pengembangan
 - Visual Paradigm sebagai tool pembuatan diagram pemodelan sistem
3. Data yang dibutuhkan meliputi:

- *Data training* yang diperoleh dari nilai mata kuliah angkatan 2007 dan 2008.
- Data masukan yaitu data nilai mata kuliah dan keminatan mahasiswa.

3.4.2 Perancangan

Setelah menentukan kebutuhan untuk membangun sistem, maka langkah selanjutnya yaitu desain atau perancangan sistem. Perancangan aplikasi berdasarkan *Object Oriented Analysis* dan *Object Oriented Design* yaitu menggunakan pemodelan UML (*Unified Modeling Language*). Pada tahap perancangan di penelitian ini, digunakan pemodelan dengan menggunakan dua macam diagram, yaitu *class diagram* dan *sequence diagram*. Perancangan kelas-kelas dan *interface-interface* yang dibutuhkan dimodelkan dalam *class diagram*. Hubungan interaksi antar elemen (objek) yang telah diidentifikasi, dimodelkan dalam *sequence diagram* yang menggambarkan interaksi antar objek yang disusun dalam urutan waktu.

3.4.3 Implementasi

Mengacu pada desain sistem, maka implementasi sistem dilakukan dengan pembangunan sistem. Implementasi perangkat lunak dilakukan dengan menggunakan bahasa pemrograman berorientasi objek yaitu menggunakan bahasa pemrograman Java (*Java Standard Edition*). Implementasi sistem pendukung keputusan dalam pemilihan konsentrasi di Program Studi Teknik Informatika meliputi:

- Pembuatan *user interface* dan penerapan algoritma *K-Nearest Neighbor* dalam program yang dibuat dengan Netbeans 6.9.
- Memasukkan data pelatihan ke *database* dengan My SQL.
- Melakukan proses perhitungan terhadap nilai yang telah dimasukkan dengan *K Nearest Neighbor*.

3.4.4 Pengujian

Pengujian perangkat lunak pada penelitian ini dilakukan agar dapat menunjukkan bahwa perangkat lunak telah mampu bekerja sesuai dengan spesifikasi dari kebutuhan yang melandasinya. Pengujian yang dilakukan meliputi pengujian performa sistem yang meliputi pengujian perangkat lunak serta pengujian validitas sistem. Strategi pengujian perangkat lunak yang digunakan yaitu pengujian unit (*unit testing*), dan pengujian validasi (*validation testing*). Sedangkan teknik atau metode pengujian yang digunakan adalah metode pengujian *white box* dan *black box*. Pada penelitian ini akan dilakukan pengujian perangkat lunak untuk sistem pendukung keputusan yang digunakan untuk pemilihan konsentrasi di Program Studi Teknik Informatika Universitas Brawijaya. Pengujian dimulai dari pengujian unit, kemudian dilanjutkan dengan pengujian integrasi, dan berakhir dengan pengujian validasi. Pada tahap pengujian unit dan pengujian integrasi digunakan metode pengujian *white box* dengan teknik *basis path*. Pada tahap pengujian validasi digunakan teknik *black-box*. Pengujian yang dilakukan yaitu pengujian *error* dan pengujian akurasi.

3.4.4.1 Pengujian Validitas Sistem

Untuk menguji validitas dari sistem, maka dilakukan uji validitas sistem terhadap data pelatihan yang telah ada. Sistem dikatakan memiliki kinerja tinggi apabila *output* yang dihasilkan oleh sistem pendukung keputusan memiliki nilai yang sama dengan *output* pada data pelatihan. Pengujian validitas sistem dihitung dengan menggunakan persamaan (2.2) dan persamaan (2.3):

3.4.5 Kesimpulan

Pembuatan kesimpulan dilakukan setelah semua tahapan perancangan, implementasi dan pengujian sistem aplikasi telah selesai dilakukan. Kesimpulan diambil dari hasil pengujian dan analisis terhadap sistem yang dibangun. Tahap terakhir dari penulisan adalah saran yang dimaksudkan untuk memperbaiki kesalahan-kesalahan yang terjadi dan menyempurnakan penulisan serta untuk memberikan pertimbangan atas pengembangan aplikasi selanjutnya.