

BAB I

PENDAHULUAN

1.1 Latar Belakang

Era digital saat ini cukup jelas membuktikan betapa pesatnya perkembangan teknologi dan informasi. Hal ini akan terus berlangsung karena semakin banyaknya minat dalam persaingan mengembangkan atau membuat berbagai teknologi baru. Pada nyatanya, bukan hanya orang yang khusus terlibat di bidang IT saja yang berpotensi dalam persaingan tersebut tetapi orang-orang di luar IT pun ikut tertarik. Oleh karena itu, pembelajaran tentang proses pembuatan perangkat lunak melalui contoh program/*coding* yang sudah ada sangat diperlukan.

Pembuatan sebuah program dapat diawali dengan mengetahui metodologi untuk memecahkan masalah yang ada kemudian menuangkan algoritma kedalam suatu notasi tertentu. Namun pada nyatanya untuk pemahaman algoritma tidak dapat disepelekan walaupun hal tersebut tidak dikatakan sulit. Hal tersebut dibuktikan dari hasil kuesioner yang dapat dilihat pada lampiran, dimana kuesioner tersebut disebarkan pada 17 orang mahasiswa jurusan Teknik Informatika di UIN Sunan Gunung Djati Bandung. Sehingga dapat diambil kesimpulan bahwa meskipun pengkodean program bukan sesuatu yang baru, namun dalam memahami dan menuangkan algoritma ke dalam kode program masih terdapat kesulitan dan masih ada yang meragukan kesulitan ketika mencari atau mengoleksi referensi apalagi jika disertai dengan membandingkan antara soal yang ingin dipecahkan dengan refensi yang banyak. Sebagian besar dari mereka

merasa perlu untuk mengetahui soal yang dipecahkan termasuk ke dalam struktur dasar algoritma yang mana dan 16 orang diantaranya merasa perlu adanya aplikasi yang dapat membandingkan kemiripan antara refensi dengan soal algoritma dan dapat memberikan rekomendasi untuk pengerjaan jawaban soal-soal algoritma. Selain itu, dari hasil peninjauan nilai akhir mahasiswa angkatan 2012-2014 pada mata perkuliahan Algoritma dan Struktur Data bahwa 28.52% diantaranya masih mempunyai hasil nilai menengah kebawah. Oleh sebab itu, pembelajaran Algoritma sangat diperlukan dalam upaya peningkatan pembelajaran agar mencapai hasil yang lebih maksimal.

Sebelum dapat memahami Algoritma perlu diketahui bahwa algoritma dibangun dari tiga struktur dasar yaitu Runtunan (*Sequence*), Pemilihan (*Selection*), dan Pengulangan (*Repetition*). Sedangkan pada sebuah kode program terdiri dari tipe data, konstanta, variabel dan algoritma dalam bentuk notasi itu sendiri. Salah satu metode yang dapat dipakai dalam konsep pembelajaran yaitu dengan menggunakan metode *Case Based Reasoning (CBR)*, karena metode ini mempunyai cara untuk memecahkan masalah yaitu dengan memanfaatkan pengalaman sebelumnya dalam domain pengetahuan tertentu, dan didukung dengan penelitian oleh Gerhana YA dan Djohar A, membuktikan bahwa metode *CBR* dapat meningkatkan kemampuan pemecahan masalah dan model pembelajaran tersebut lebih baik dari model konvensional [1]. Dalam penelitian lain dilakukan oleh Luthfi ET, mengungkapkan bahwa penggunaan *CBR* dapat dilakukan untuk mencari level kedekatan data kasus baru dengan data kasus lama yang menjadi acuan dalam pengambilan keputusan terhadap kasus baru [2].

Perhitungan tingkat kemiripan (jarak) didasarkan pada penggunaan beberapa atribut yang terdefinisi sebelumnya dan dapat menggunakan algoritma pengklasifikasi seperti *K-Nearest Neighbor*, *Naive bayes*, serta gabungan *LVQ* dan *K-Means*. Seperti pada hasil penelitian yang telah dilakukan sebelumnya oleh Santoso D, Ratnawati DE dan Indriati bahwa rata-rata akurasi *KNN* 96%, *Naive Bayes* 98%, serta gabungan *k-means* dan *LVQ* 92,2% [3]. Pada keempat algoritma tersebut dikatakan baik untuk digunakan terutama algoritma *K-Nearest Neighbor* (*KNN*) yang diperkirakan akan menghasilkan pengklasifikasian obyek baru yang lebih cocok untuk dijadikan bahan penelitian karena *KNN* berdasar pada atribut dan data *training* atau mengklasifikasikan dengan *voting* terbanyak dari data yang telah ada sebelumnya, dan apabila menggunakan tambahan metode lain seperti *Decision Rule* dalam penelitian oleh Samuel Y, Delima R dan Rachmat A mempunyai hasil kurang mampu memaksimalkan performa *KNN* [4], sedangkan dengan *Improved KNN* pada penelitian oleh Rizki AS, Indriati, dan Muflikhah menghasilkan performa yang kurang bagus [5].

Pencarian kemiripan soal dan jawaban algoritma dengan menggunakan algoritma klasifikasi merupakan sebuah solusi dalam memecahkan permasalahan dari soal algoritma. Untuk itu, maka disusunlah penelitian “**Implementasi Algoritma *K-Nearest Neighbor* dalam Sistem *Case Based Reasoning* untuk Pembentukan Identitas Jawaban Otomatis dan Pencari Kemiripan Jawaban dari Soal-Soal Algoritma**”.

1.2 Rumusan Masalah

Rumusan masalah dari latar belakang diatas yang akan dijadikan objek pada penelitian adalah sebagai berikut:

1. Bagaimana hasil kolaborasi antara sistem *Case Based Reasoning* dan algoritma *K-Nearest Neighbor* dalam membentuk identitas jawaban otomatis dari soal algoritma dan mencari kemiripan jawaban tersebut?
2. Bagaimana penerapan algoritma *K-Nearest Neighbor* dan metode *Case Based Reasoning* untuk mengetahui tingkat akurasi hasil pembentukan identitas jawaban otomatis dan pencari kemiripan jawaban?

1.3 Tujuan

Tujuan yang ingin dicapai dari permasalahan pada penelitian ini yakni sebagai berikut:

1. Mengetahui hasil kolaborasi dari sistem *Case Based Reasoning* dan algoritma *K-Nearest Neighbor* dalam membentuk identitas jawaban otomatis dari soal algoritma dan mencari kemiripan jawaban.
2. Menerapkan algoritma *K-Nearest Neighbor* dan metode *Case Based Reasoning* untuk mengetahui tingkat akurasi hasil pembentukan identitas jawaban otomatis dan pencari kemiripan jawaban.

1.4 Batasan Masalah

Batasan permasalahan dari penelitian berguna agar penulisan penelitian lebih teratur dan terarah. Adapun batasan permasalahan tersebut meliputi:

1. Implementasi dari sistem ini berbasis *desktop offline*.

2. Pembentukan identitas jawaban otomatis diperoleh dari perbandingan kemiripan antara soal inputan yang ingin dicarikan jawaban dengan soal dalam pengetahuan yang telah memiliki identitas jawaban sebelumnya.
3. Jawaban hasil pencarian berupa 5 rekomendasi yang diurutkan berdasarkan tingkat kemiripan tertinggi dan klasifikasi dari *vote* terbanyak sehingga dapat dijadikan sebagai gambaran untuk pengerjaan soal-soal algoritma yang jawabannya harus menggunakan *coding*.
4. Potongan kode program berupa gambar dalam bahasa pemrograman C, Pascal dan C++.
5. Tingkat akurasi diperoleh dari hasil pengklasifikasian terhadap pengujian sistem menggunakan Algoritma *K-Nearest Neighbor* dalam sistem *Case Based Reasoning*.

1.5 Metodologi Penelitian

Metodologi yang digunakan pada penelitian ini terbagi menjadi 2 bagian yaitu teknik pengumpulan data dan model proses pengembangan perangkat lunak.

1.5.1 Teknik Pengumpulan Data

Metodologi Penelitian diperlukan sebagai alat bantu untuk memudahkan pekerjaan didalam melakukan perancangan aplikasi. Metodologi penelitian yang digunakan yaitu sebagai berikut:

a. Studi Kepustakaan

Studi pustaka merupakan langkah awal dalam metode pengumpulan data.

Studi pustaka merupakan metode pengumpulan data yang diarahkan kepada pencarian data dan informasi melalui jurnal penelitian, internet, buku dan e-

book yang dapat mendukung dalam proses penulisan. "Hasil penelitian juga akan semakin kredibel apabila didukung foto-foto atau karya tulis akademik dan seni yang telah ada [6]." Maka dapat dikatakan bahwa studi pustaka dapat memengaruhi kredibilitas hasil penelitian yang dilakukan.

b. Wawancara

Wawancara atau *interview* merupakan teknik pengumpulan data dengan cara bertatap muka secara langsung dengan informan dan dosen yang bersangkutan. Wawancara dilakukan jika data yang diperoleh kurang mendalam. Wawancara digunakan sebagai teknik pengumpulan data apabila peneliti ingin mengetahui hal-hal dari informan yang lebih mendalam [6].

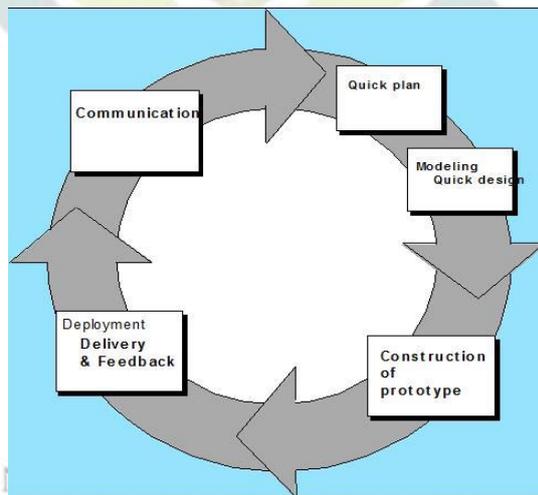
1.5.2 Model Proses Pengembangan Perangkat Lunak

Adapun untuk pembuatan aplikasi ini, menggunakan metode pengembangan perangkat lunak *Prototype*, karena metode ini lebih memudahkan dalam proses membangun sebuah perancangan aplikasi. Alur dari metode *Prototype* terdiri dari langkah-langkah sebagai berikut:

- a. *Communication*, yaitu *Developer* dan klien bertemu dan menentukan tujuan umum, kebutuhan yang diinginkan dan gambaran bagian-bagian yang akan dibutuhkan berikutnya.
- b. *Quick Plan* atau Perancangan yang dilakukan dengan cepat dan mewakili semua aspek *software* yang diketahui, dan rancangan ini menjadi dasar pembuatan *Prototype*.
- c. *Modelling Quick Design*, langkah ini berfokus pada representasi aspek *software* yang bisa dilihat *customer/User* dan cenderung ke pembuatan *prototype*.

- d. *Construction of Prototype*, yaitu membangun kerangka atau rancangan *prototype* dari *software* yang akan dibangun.
- e. *Deployment Delivery & Feedback* merupakan *Prototype* yang telah dibuat oleh *developer* akan disebarkan kepada *User/klien* untuk dievaluasi, kemudian klien akan memberikan *feedback* yang akan digunakan untuk merevisi kebutuhan *software* yang akan dibangun. Pengulangan proses ini terus berlangsung sampai semua kebutuhan terpenuhi [7].

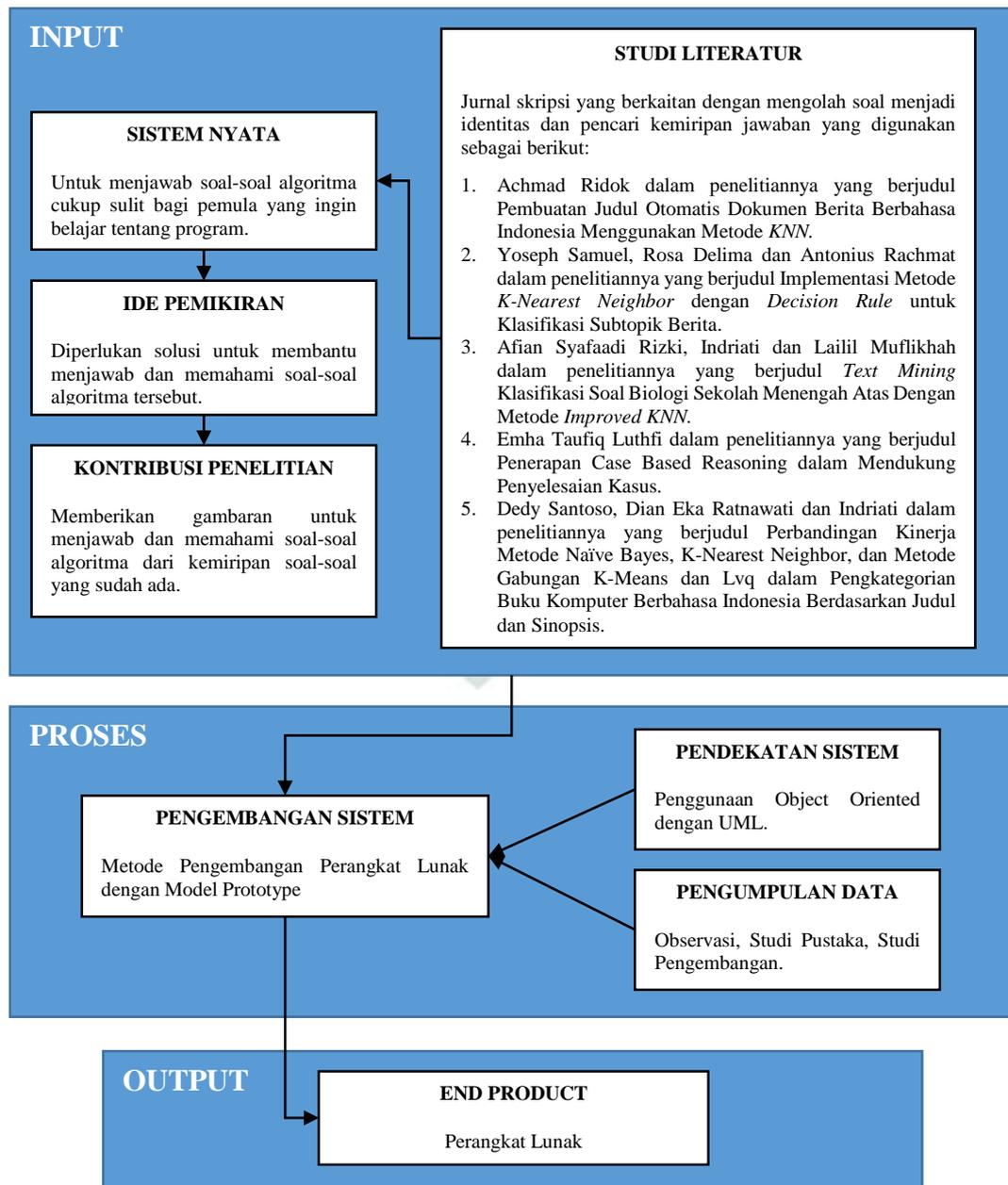
Alur tersebut dapat dijelaskan pada gambar ilustrasi proses model *prototype* sebagai berikut:



Gambar 1.1 Alur Model Pengembangan *Prototype* [7]

1.6 Kerangka Kerja Konseptual

Kerangka kerja konseptual atau alur logis dari penelitian yang akan dibuat adalah sebagai berikut:



Gambar 2.2 Skema Kerangka Kerja Konsep

Dilihat dari kerangka kerja konsep diatas terbagi menjadi 3 tahapan yang bertujuan untuk memudahkan dalam urutan perjabaran dan klasifikasi pencapaian dari skema tersebut. Tahapan tersebut terdiri dari:

1.6.1 *Input*

Tahapan awal yaitu input yang akan menjadi cangkupan pada beberapa kegiatan yang dilakukan seperti:

a. Studi Literatur

Studi Literatur adalah cara untuk menyelesaikan persoalan dengan menelusuri sumber-sumber tulisan yang pernah dibuat sebelumnya. Studi literatur yang dilakukan adalah dengan mencari data dan informasi yang dibutuhkan. Data dan informasi yang dibutuhkan dalam penelitian ini dicari melalui jurnal-jurnal ilmiah terakreditasi, dan hasil-hasil penelitian skripsi, tesis, disertasi, dan laporan praktikum yang sebelumnya pernah dijadikan bahan penelitian.

b. Sistem Nyata

Setelah studi literatur dilakukan, maka didapat permasalahan nyata yang saat ini masih dijadikan perbincangan dan belum ditemukan solusi yang lebih baik. Pada sistem nyata penelitian ini ditemukan permasalahan dalam menjawab soal-soal algoritma cukup sulit bagi pemula yang ingin belajar tentang program.

c. Ide Pemikiran

Setelah ditemukan permasalahan pada sistem nyata munculah ide pemikiran mengenai perlunya solusi untuk membantu menjawab dan memahami soal-soal algoritma.

d. Kontribusi Penelitian

Untuk mendukung solusi permasalahan dari ide yang didapatkan, diperlukan kontribusi penelitian yang berguna dalam memberikan gambaran untuk menjawab dan memahami soal-soal algoritma dari kemiripan soal-soal yang sudah ada.

1.6.2 Proses

Proses merupakan tahapan yang selanjutnya yang dilakukan setelah tahap input dan sebelum tahap output. Pengertian dari proses itu sendiri yaitu berupa serangkaian kegiatan yang saling terkait atau berinteraksi yang nantinya akan mengubah input menjadi output. Adapun tahapan proses mencakup pada beberapa kegiatan yakni sebagai berikut:

a. Pendekatan Sistem

Pendekatan sistem dilakukan dengan membuat perancangan terhadap sistem dengan menggunakan metode pengembangan sistem *Unified Modelling Language* (UML).

b. Pengumpulan Data

Pengumpulan data dilakukan dengan melakukan Studi kepustakaan dan wawancara. Studi pustaka merupakan metode pengumpulan data yang diarahkan kepada pencarian data dan informasi melalui jurnal penelitian, internet, buku dan *e-book* yang dapat mendukung dalam proses penulisan. Sedangkan wawancara dilakukan dengan cara bertatap muka secara langsung dengan informan atau dosen yang bersangkutan. Wawancara digunakan sebagai teknik pengumpulan data apabila peneliti ingin mengetahui berbagai informasi dari informan dengan lebih mendalam.

c. Pengembangan Sistem

Setelah pendekatan sistem dan pengumpulan data selesai, barulah dapat dilakukan pengembangan terhadap sistem. Pengembangan sistem untuk penelitian ini akan menggunakan metode pengembangan perangkat lunak dengan model *prototype*, karena metode ini dianggap lebih memudahkan dalam proses membangun sebuah perancangan aplikasi.

1. *Communication*

Developer dan klien bertemu dan menentukan tujuan umum, kebutuhan yang diinginkan dan gambaran bagian-bagian yang akan dibutuhkan berikutnya.

2. *Quick Plan*

Perancangan dilakukan cepat dan mewakili semua aspek *software* yang diketahui, dan rancangan ini menjadi dasar pembuatan *Prototype*.

3. *Modelling Quick Design*

Berfokus pada representasi aspek *software* yang bisa dilihat *customer/User*. *Modelling QuickDesign* cenderung ke pembuatan *prototype*.

4. *Construction of Prototype*

Membangun kerangka atau rancangan *Prototype* dari *software* yang akan dibangun.

5. *Deployment Delivery & Feedback*

Prototype yang telah dibuat oleh *developer* akan disebarkan kepada *User/klien* untuk dievaluasi, kemudian klien akan memberikan *feedback* yang akan digunakan untuk merevisi kebutuhan *software* yang

akan dibangun. Pengulangan proses ini terus berlangsung sampai semua kebutuhan terpenuhi.

1.6.3 Output

Tahapan yang terakhir adalah tahap *output*, yang dimana merupakan hasil akhir keputusan yang dipertimbangkan dari pemikiran kerangka kerja konsep. Tahapan ini mencangkupi hasil yang berupa perangkat lunak.

1.7 Sistematika Penulisan

Sistematika penulisan laporan tugas akhir disusun dalam beberapa bab yang masing-masing bab menguraikan beberapa pokok pembahasan. Adapun sistematika penulisan laporan ini adalah sebagai berikut:

BAB 1 PENDAHULUAN

Bab pendahuluan terdiri dari enam subbab yaitu latar belakang menggambarkan hal-hal yang perlu dijadikan penelitian dari realita yang ada, rumusan masalah, tujuan, batasan masalah, metodologi penelitian, kerangka kerja konsep dan sistematika penulisan. Subbab Latar belakang masalah menggambarkan hal-hal yang perlu dijadikan penelitian dari realita yang ada. Subbab Rumusan masalah dituliskan kedalam bentuk poin yang menjadi sasaran utama pada objek yang akan diteliti. Subbab Tujuan menggambarkan hal-hal yang ingin dicapai. Subbab Batasan masalah berisi batasan yang ditentukan dalam perancangan sistem berupa hal-hal terkait dengan sistem. Subbab Kerangka Kerja Konsep menggambarkan keseluruhan konsep umum dan ide dari penelitian. Subbab Metodologi penelitian terbagi kedalam dua tahapan yaitu teknik pengumpulan data dan menggambarkan model proses pengembangan perangkat lunak yang dibuat.

Dan bagian terakhir dari bab ini yakni dengan subbab sistematika penulisan yang menguraikan urutan penyajian yang digunakan dalam penyusunan skripsi.

BAB II LANDASAN TEORI

Bab landasan teori menjelaskan tentang teori-teori yang digunakan dalam menganalisis permasalahan yang ada dan teori-teori yang akan digunakan untuk membangun perangkat lunak ini.

BAB III ANALISIS DAN PERANCANGAN

Bab analisis dan perancangan menguraikan hasil analisis dan perancangan perangkat lunak yang akan dibangun.

BAB IV IMPLEMENTASI

Bab implementasi menjelaskan tentang spesifikasi perangkat lunak, kebutuhan perangkat lunak, implementasi perangkat lunak, dan pengujian yang dilakukan terhadap perangkat lunak yang dibangun.

BAB V PENUTUP

Bab penutup berisi tentang pernyataan singkat berupa kesimpulan dari pembahasan perangkat lunak yang dibuat secara keseluruhan dan saran untuk mengembangkan perangkat lunak yang lebih baik.



uin

UNIVERSITAS ISLAM NEGERI
SUNAN GUNUNG DJATI
BANDUNG