

1. Pendahuluan

1.1 Latar belakang

PT. Kereta Api merupakan satu-satunya perusahaan negara yang bergerak dibidang jasa transportasi kereta api. Dalam proses bisnisnya, ada sebuah bagian yang bertugas untuk memantau trafik kereta api setiap harinya. Bagian ini terdistribusi di setiap station yang dilalui oleh kereta api. Antara stasiun satu dengan yang lain yang terhubung oleh satu jalur kereta api saling berkomunikasi lewat radio untuk bertukar informasi kedatangan dan keberangkatan kereta di station yang dilaluinya dengan station yang sedang ditujunya.

Informasi dari stasiun asal inilah yang dijadikan rujukan untuk memperkirakan kedatangan kereta api. Sehingga kevalidan informasi menjadi sesuatu yang sangat penting. Ketidakvalidan informasi yang didapat akan sangat mempengaruhi hasil prediksi yang dihasilkan. Hasil prediksi inilah yang akan dijadikan bahan pertimbangan untuk pengambilan keputusan apakah kereta tersebut akan diteruskan ataukah menunggu sementara waktu jika pada saat yang bersamaan ada kereta api yang berjalan berlawanan menuju stasiun yang sama.

Saat ini proses kerja seperti ini sebagian masih dilakukan dengan cara konvensional. Petugas CTC meminta informasi dari station asal kereta lewat radio. Kemudian informasi tersebut digunakan untuk memprediksi secara manual kedatangan kereta api pada stasiun tersebut dengan menggunakan kertas yang berisi grafik xy dimana x mewakili waktu dan y mewakili stasiun-stasiun yang dilalui kereta api tersebut.

Dari gambaran proses kerja di atas, maka dapat dibayangkan besar resiko yang akan dialami jika terjadi kesalahan perhitungan dalam memprediksi kedatangan kereta api tersebut. Kesalahan-kesalahan tersebut dapat disebabkan oleh kurang telitinya petugas dalam melakukan perhitungan karena faktor manusiawi.

Bidirectional search merupakan algoritma pencarian yang memanfaatkan dua *node* ujung untuk mencari *node* yang berada di antara dua *node* tersebut. *Bidirectional search* efisien jika *node* yang digunakan jumlahnya terbatas. Oleh karena itu *bidirectional search* sesuai untuk menentukan titik pertemuan dua kereta api yang berjalan berlawanan arah pada jalur yang sama. Dengan mengetahui titik pertemuan tersebut maka pengaturan trafik kereta api dan informasi bahwa jalur antar stasiun dalam kondisi aman dapat ditentukan dengan lebih mudah.

1.2 Perumusan masalah

Permasalahan yang dijadikan objek penelitian dan pengembangan tugas akhir ini adalah bagaimana menentukan titik temu yang aman antara dua kereta api yang melintasi jalur rel *single track* dan bagaimana penggambaran trafik perjalanan dalam koordinat XY (waktu, jarak/stasiun) serta simulasi perjalanan kereta api. Untuk menghindari meluasnya materi pembahasan tugas akhir ini, maka penulis membatasi permasalahan dalam tugas akhir ini sebagai berikut :

1. Sistem ini digunakan untuk menangani trafik kereta api yang menggunakan jalur rel *single track* dan *double track*..
2. Sistem yang dihasilkan akan digunakan untuk trafik kereta api jurusan Cikampek – Bandung.

3. *Bidirectional search* hanya digunakan untuk mencari titik pertemuan pada penyilangan dua kereta api di jalur *single track*.
3. Variabel yang digunakan dalam pemantauan adalah jadwal perjalanan kereta api dan waktu tempuh kereta api antar stasiun.

1.3 Tujuan

Tujuan dari Tugas Akhir ini adalah membangun dan mengembangkan sebuah sistem pemantau kereta api, di mana sistem ini dapat memberikan hasil perkiraan trafik kereta api berdasarkan jadwal keberangkatan kereta api, mampu melakukan perbaikan terhadap perkiraan trafik yang salah terhadap keadaan nyata, mampu menggambarkan simulasi perjalanan kereta api terhadap hasil perkiraan tersebut dan menganalisa sistem yang dibangun.

1.4 Metodologi penyelesaian masalah

Perangkat lunak sistem ini dibangun dengan tahapan :

1. Tahap Studi Literatur
Melalui metode ini dilakukan tinjauan beberapa aspek pendukung seperti buku, situs web, jurnal, majalah, dan artikel yang berkaitan dengan topik ini.
2. Tahap Analisis dan Definisi Kebutuhan
Mempelajari sistem yang telah ada sehingga bisa dibangun aplikasi baru yang dapat memenuhi kebutuhan *user*. Mengumpulkan data-data, baik mekanisme pemantauan secara manual yang sedang berjalan, sehingga dapat dibuat model yang sesuai.
3. Tahap Perancangan Sistem dan Perangkat Lunak
Merancang suatu sistem aplikasi baru yang dapat memenuhi kebutuhan *user* secara optimal. Perancangan perangkat lunak termasuk mempresentasikan fungsi-fungsi sistem perangkat lunak, sehingga dapat ditransformasikan kedalam sebuah "*Executable Program*".
4. Tahap Implementasi
Selama tahap ini perancangan perangkat lunak direalisasikan kedalam kumpulan program atau unit-unit program, termasuk pengujian terhadap unit-unit tersebut sesuai dengan spesifikasinya.
5. Tahap Integrasi dan Pengujian Sistem
Tahap ini unit-unit program diintegrasikan dan diuji sebagai suatu sistem secara keseluruhan, untuk memastikan bahwa sistem sudah sesuai dengan spesifikasinya.