

# Sistem Konektivitas PcVue ke SQL dan Python

1<sup>st</sup> Havis Eka Permana  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

haviseka@student.telkomuniversity.ac.id

2<sup>nd</sup> Ahmad Sugiana  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

sugianaa@telkomuniversity.ac.id

3<sup>rd</sup> Junarto Halomoan  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

junartha@telkomuniversity.ac.id

**Abstrak** — Penelitian ini bertujuan untuk mempermudah pengawasan *Central Traffic Control* (CTC) dengan mengintegrasikan database PcVue ke dalam *SQL Server Management Studio* (SSMS). Proses integrasi ini memungkinkan perubahan data di PcVue secara otomatis tercatat di SSMS, sehingga memudahkan pemantauan secara langsung tanpa perlu berada di ruangan CTC. Penelitian ini melibatkan pembuatan server di PcVue dan penghubungannya ke SQL Server, serta pengujian durasi waktu yang dibutuhkan untuk sinkronisasi data antara PcVue dan SQL. Hasil pengujian menunjukkan bahwa setiap perubahan data di PcVue membutuhkan waktu satu detik untuk tercatat di SSMS, menunjukkan efisiensi sistem dalam sinkronisasi data.

**Kata kunci**—*Central Traffic Control* (CTC), *PcVue*, *SQL Server Management Studio* (SSMS), *Integrasi Sistem*, *Sinkronisasi Data*

## I. PENDAHULUAN

Persinyalan kereta api merupakan suatu bentuk, warna, atau cahaya yang memberikan isyarat kepada petugas perkeretaapian untuk memastikan perjalanan kereta api yang aman [1]. Jenis persinyalan kereta api terbagi menjadi dua, yaitu persinyalan elektrik dan mekanik. Persinyalan elektrik adalah peralatan yang memberikan isyarat berupa cahaya atau warna yang memiliki arti tertentu dan biasanya berupa lampu LED (Light Emitting Diode) [2]. Sebaliknya, sinyal mekanik adalah perangkat sinyal yang digerakkan secara mekanik, contohnya papan atau lengan instruksi yang dinaikkan dan diturunkan untuk memberi perintah kepada masinis kereta api [3].

Kondisi persinyalan kereta api dapat dipantau melalui CTC (Centralized Traffic Control), yaitu sebuah sistem monitoring dan kontrol persinyalan kereta api yang berasal dari Amerika Utara. CTC memungkinkan pemantauan kondisi persinyalan kereta api apakah berjalan normal atau sedang terjadi gangguan, yang kemudian dipantau oleh operator dan teknisi pemeliharaan di ruang kontrol [4].

PcVue adalah perangkat lunak SCADA (Supervisory Control and Data Acquisition) yang mengelola data dan bit value dari persinyalan kereta api. Dalam PcVue, nilai bit 0 menunjukkan bahwa persinyalan dalam kondisi normal, sedangkan nilai bit 1 menunjukkan adanya gangguan pada persinyalan. Sistem yang akan dibuat bertujuan untuk mengawasi persinyalan kereta api yang berada pada CTC.

Fokus utama dari jurnal ini adalah membahas proses transfer data yang terjadi dari PcVue, perangkat lunak yang digunakan sebagai SCADA, yang akan diawasi dari jarak jauh melalui perangkat smartphone.

Transfer data ini memerlukan perangkat lunak pendukung seperti *SQL Server Management Studio* (SSMS) dan skrip bahasa pemrograman Python untuk mengirim data tersebut ke cloud yang disediakan. Data dari PcVue akan dikirimkan ke *SQL Server Management Studio* (SSMS), yang merupakan database SQL. Data SQL ini kemudian diubah menjadi format JSON menggunakan skrip Python agar dapat diterima oleh Firebase, karena Firebase tidak dapat menerima data SQL secara langsung.

## II. KAJIAN TEORI

### A. PcVue

PcVue terdiri atas Field Device Layer, Communication Layer, Server Layer, Application Layer. Pada Field Device Layer terdapat sensor, aktuator, PLC, RTU untuk mengumpulkan data di lapangan kemudian layer ini dibantu oleh Communication Layer untuk mengirimkan datanya ke PcVue sebagai Server SCADA yang terdapat di Server Layer. PcVue berfungsi sebagai server SCADA untuk mengumpulkan, memproses, mengontrol, dan menyimpan data. Pada Application Layer berisi Human Machine Interface (HMI) sebagai grafis antarmuka bagi operator untuk memonitor, mengontrol, dan menganalisis.

### B. SQL Server Management Studio

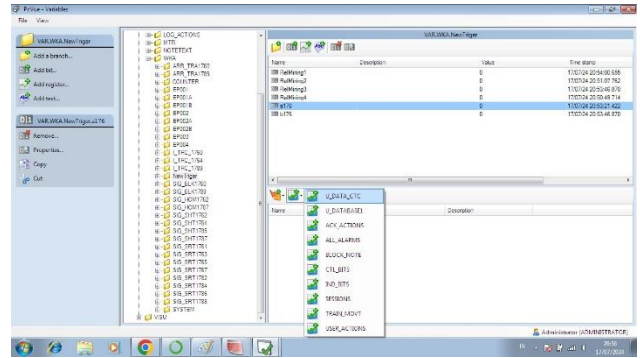
SQL terdiri atas Protokol Layer, Relational Engine, dan Storage Engine. Protocol Layer adalah lapisan yang bertanggung jawab untuk mengatur komunikasi antara client dan server database. Relational Engine adalah yang bertanggung jawab untuk memproses dan menjalankan query SQL. Query SQL merupakan perintah yang berhubungan dengan database melalui SQL. SQL dapat digunakan untuk mengambil, memasukkan, memperbarui, menghapus data, dan mengatur struktur database.[5] Storage Engine adalah komponen untuk penyimpanan data. Plan Cache adalah untuk meningkatkan kinerja Relational Engine agar dapat menyimpan plan eksekusi query yang sering digunakan sehingga dimasa depan query yang sama hanya perlu dijalankan tanpa perlu dioptimalkan ulang.[6]

### III. METODE

Penelitian ini bertujuan untuk mempermudah pengawasan CTC, sehingga pengawasan tidak diharuskan didalam ruangan CTC.

#### A. Mempersiapkan Database PcVue

Database PcVue agar bisa dikirim ke SQL Server Management Studio maka harus melewati beberapa langkah yang di mana adanya pembuatan server yang harus dihubungkan dengan SQL Server agar bit yang berubah dalam PcVue dapat langsung terkirim ke SQL Server Management Studio.

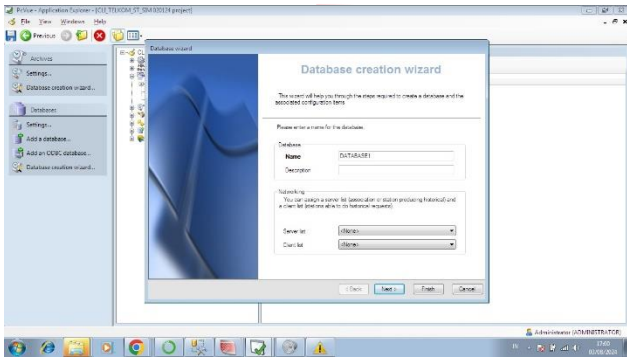


Gambar3.3

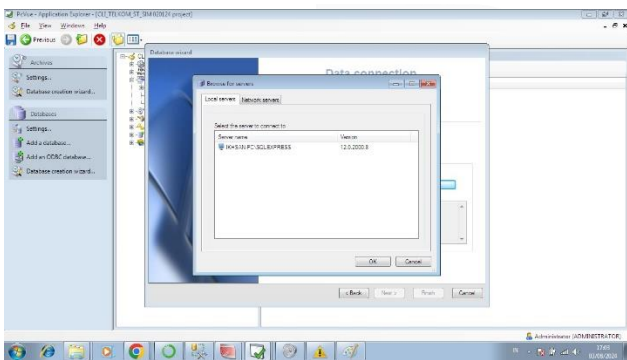
Gambar 3.3 merupakan salah satu bagian proses penambahan variable kedalam database yang tadi telah dibuat dan terhubung ke SQL Server Management Studio.

#### B. SQL Server Management Studio (SSMS)

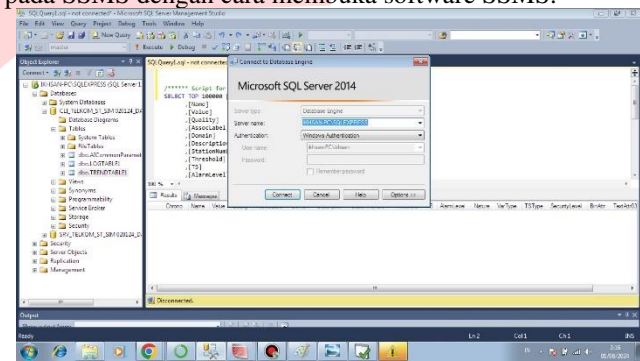
Memeriksa hasil yang telah di tambahkan dari PcVue pada SSMS dengan cara membuka software SSMS.



GAMBAR3.1



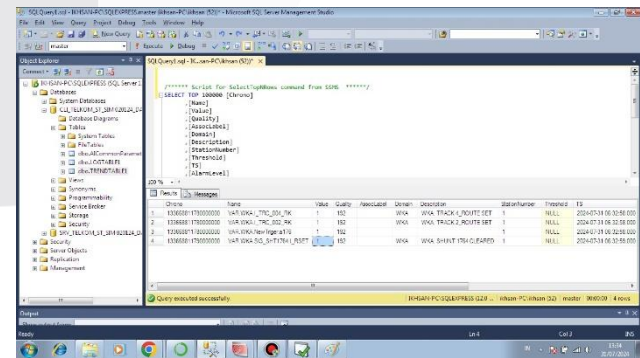
GAMBAR3.2



GAMBAR3.4

Gambar 3.4 merupakan tampilan awal SSMS yang dimana kita harus menghubungkan terlebih dahulu ke server agar pembacaan database yang berhasil kita buat dan hubungkan serta tambahkan dari PcVue dapat terbaca oleh software SSMS. Setelah Terhubung maka tampilannya seperti gambar di bawah ini.

Gambar 3.1 adalah proses awal pembuatan server di PcVue yang dimana proses awal tersebut adalah proses penamaan server. Gambar 3.2 adalah proses pemilihan server SQL Server yang dimana SQL Server sendiri sudah ada saat penginstalan SQL Server Management Studio. Setelah menyelesaikan hal tersebut maka dilanjutkan proses penambahan bit dari PcVue ke database yang telah terhubung dengan SQL Server tersebut.



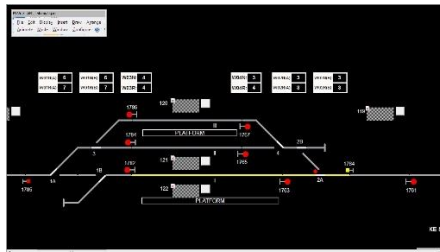
GAMBAR3.5

### IV. HASIL DAN PEMBAHASAN

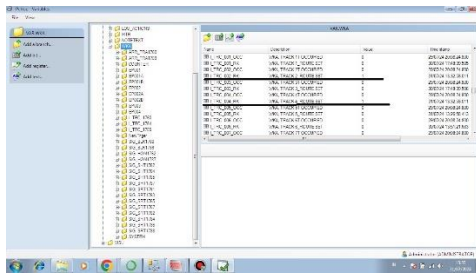
#### A. Pengecekan data PcVue ke SQL

1. Membukak software PcVue
2. Menunjukkan track simulasi yang bermasalah pada PcVue

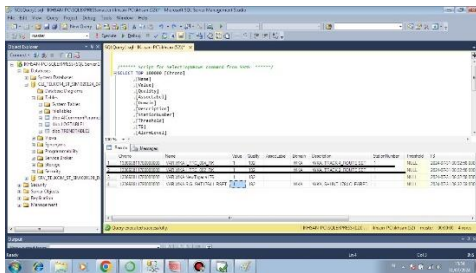
3. Membuka software SSMS
4. Menghubungkan Server SQL
5. Pencocokan bit yang telah berubah



GAMBAR 4.1



GAMBAR 4.2



GAMBAR 4.3

Gambar 4.1 merupakan gambar contoh track simulasi yang sedang memiliki masalah. Pada gambar 4.2 merupakan variabel bit yang berada pada software PcVue. Pada gambar tersebut dapat di lihat bawahannya TRC 02 RK dan TRC 04 RK yang merupakan nama track tersebut saat ada gangguan maka vauluenya akan berubah menjadi 1. Untuk gambar 4.3 adalah gambar tampilan SSMS yang dimana tertera kedua variable track yang berada pada PcVue tadi telah tercatat pada SSMS yang dimana TRC 02 RK dan TRC 04 RK telah memiliki value 1.

1. Analisis Pengujian

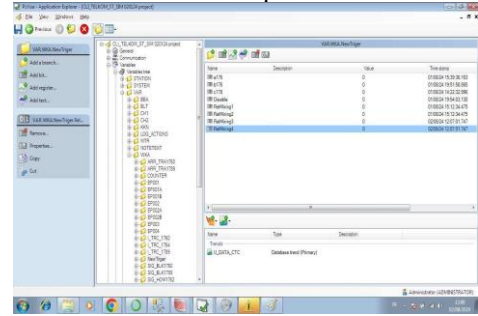
Hasil dari pengujian ini menunjukkan bahwa data di PcVue telah terekam dengan benar di SSMS, mempermudah penanganan dan analisis masalah track. *Verifikasi* ini penting untuk memastikan integritas data antara sistem PcVue dan *database* SQL, serta membantu pemantauan dan penyelesaian masalah secara efisien.

B. Hasil Pengujian waktu

Langkah pengujian

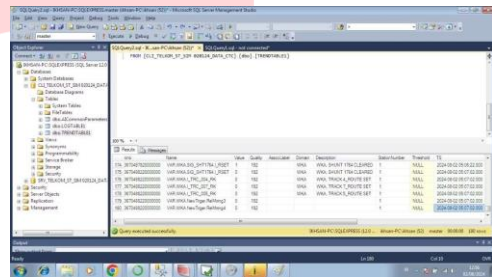
1. Buka PcVue lalu ubah persinyalan
2. masuk ke bagian 'Variable Selector' pada PcVue untuk melihat timestamp perubahan sinyal
3. Buka SSMS lalu 'Execute' program untuk memperbarui row pada SQL
4. Catat nilai timestamp pada PcVue, dan SMSS

5. Rata-rata kan durasi waktu perubahan data



GAMBAR 4.4

Pada Gambar 4.4 merupakan tampilan dari bit *value* yang akan berubah saat ada perubahan kondisi persinyalan serta *timestamp* yang mengindikasi waktu perubahan *value*, *value* ini terdiri atas 2 bit yaitu value 0 dan 1 yang berarti kondisi normal dan gangguan. Pada pengujian ini bit RelMiring4 diubah *value* nya menjadi 0 di waktu 12:07:01 yang terampil pada kolom *timestamp*.



GAMBAR 4.5

Gambar 4.5 merupakan tampilan pada SSMS yang menampilkan data PcVue yang diubah menjadi SQL, pada SSMS terdapat *timestamp* yang diberi nama 'TS', pada kolom ini merupakan waktu saat data PcVue berhasil diterima di SSMS namun terdapat kekurangan dimana waktu yang tertampil memiliki selisih 7 jam namun menit dan detik masih sesuai. Pada pengujian RelMiring4 ditampilkan data diterima pada waktu 05:07:02, agar sesuai dengan waktu seharusnya maka ditambahkan 7 jam menjadi 12:07:02 yang berarti membutuhkan waktu 1 detik untuk mengirimkan data dari PcVue sampai diterima di SSMS.

Percobaan	PcVue	SQL	Waktu PcVue ke SQL
1	11:39:17	11:39:18	00:00:01
2	11:41:26	11:41:27	00:00:01
3	11:43:20	11:43:21	00:00:01
4	11:45:00	11:45:01	00:00:01
5	11:46:57	11:46:58	00:00:01
6	11:49:05	11:49:06	00:00:01
7	11:50:02	11:50:02	00:00:00
8	11:50:58	11:50:59	00:00:01
9	11:51:37	11:51:38	00:00:01
10	11:52:27	11:52:28	00:00:01
11	11:53:22	11:53:23	00:00:01
12	11:54:12	11:54:13	00:00:01
13	11:54:55	11:54:56	00:00:01
14	11:55:43	11:55:43	00:00:00
15	11:56:29	11:56:30	00:00:01
16	11:57:17	11:57:18	00:00:01

17	11:58:03	11:58:04	00:00:01
18	11:58:47	11:58:48	00:00:01
19	11:59:33	11:59:34	00:00:01
20	12:00:23	12:00:24	00:00:01
21	12:01:07	12:01:07	00:00:00
22	12:01:43	12:01:44	00:00:01
23	12:02:18	12:02:19	00:00:01
24	12:02:48	12:02:49	00:00:01
25	12:03:33	12:03:34	00:00:01
26	12:04:10	12:04:11	00:00:01
27	12:04:51	12:04:52	00:00:01
28	12:05:27	12:05:28	00:00:01
29	12:06:20	12:06:21	00:00:01
30	12:07:01	12:07:02	00:00:01
Rata-Rata			00:00:01

GAMBAR  
Tabel Percobaan

- [2] L. Nawangwulan, "Persinyalan," 2021.
- [3] E. Sayuri1, "Perancangan Sistem Persinyalan Elektrik Di Stasiun Berbasis PLC Omron," 2017.
- [4] J. Alvert, Controlize Traffic Control, 1999.
- [5] T. Putra, "Perancangan Aplikasi Dashboard Untuk Monitoring," 2018.
- [6] S. YAHYA, "Rancang Bangun Sistem Lelang Online," 2010.

## 2. Analisis Pengujian

Berdasarkan tabel yang disajikan, analisis waktu antara perubahan bit di PcVue dan pencatatannya di SQL menunjukkan hasil yang signifikan. Pada setiap percobaan, dicatat waktu ketika perubahan terjadi di PcVue dan waktu ketika perubahan tersebut tercatat di SQL. Selisih waktu antara kedua pencatatan ini secara konsisten adalah satu detik. Kolom "Waktu PcVue ke SQL" memperlihatkan selisih waktu antara pencatatan di PcVue dan SQL, yang secara konsisten menunjukkan angka 00:00:01 (satu detik) untuk setiap percobaan. Hal ini menunjukkan bahwa sistem memerlukan waktu satu detik untuk menyinkronkan perubahan data dari PcVue ke SQL.

## V. KESIMPULAN

Berdasarkan hasil pengujian dan analisis, sistem yang dikembangkan mampu menyinkronkan perubahan data dari PcVue ke SQL Server Management Studio (SSMS) dengan waktu yang konsisten, yaitu satu detik. Hal ini membuktikan bahwa integrasi antara PcVue dan SSMS berjalan dengan baik dan efisien, memudahkan pengawasan dan pemantauan CTC secara langsung. Keberhasilan ini juga menunjukkan bahwa sistem mampu menjaga integritas data selama proses *transfer*, sehingga dapat diandalkan untuk penggunaan operasional sehari-hari di CTC.

## REFERENSI

- [1] D. S. O. A. Darmawan, "Peningkatan Keamanan Perjalanan Kereta Api Dengan Penggunaan Sistem Axle Counter Dan Media Transmisi Fiber Optic Untuk Hubungan Blok Di Persinyalan Vpi (Studi Kasus Hubungan Blok Stasiun Surodadi – Pematang)," 2017.