Jobsheet 8

EMAIL PROTOKOL



Oleh :

Nama : Maharani Safitri

NIM : 21343008

PROGRAM STUDI INFORMATIKA (NK)

JURUSAN TEKNIK ELEKTRONIKA

FAKULTAS TEKNIK

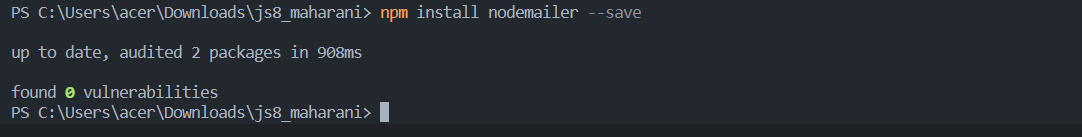
UNIVERSITAS NEGERI PADANG

TAHUN 2023

1. Latar belakang:

Latar belakang dari aplikasi ini adalah untuk memberikan panduan atau contoh penggunaan modul Nodemailer dalam Node.js untuk mengirimkan email melalui layanan Gmail. Aplikasi ini tampaknya bertujuan sebagai tutorial atau pembelajaran bagi pengembang yang ingin memahami cara menggunakan Nodemailer untuk mengintegrasikan kemampuan pengiriman email dalam aplikasi Node.js mereka.

1. LANGKAH LANGKAH
2. **Buka Teks Editor:**
   * Buka teks editor di komputer .
3. **Pastikan Nodemailer Terinstal:**
   * Pastikan Node.js sudah terinstal di komputer .
   * Pastikan Nodemailer sudah terinstal dengan menjalankan perintah instalasi: **npm install nodemailer**.

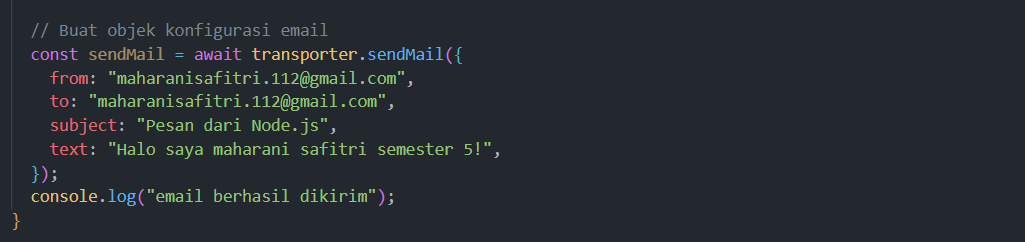


1. **Buat Fungsi Pengiriman Email:**
   * Buat fungsi untuk mengirim email. Beri nama fungsi ini **kirimEmail**.
2. **Buat Objek Transporter:**
   * Dalam fungsi **kirimEmail**, buat objek **Transporter** menggunakan **nodemailer.createTransport**.

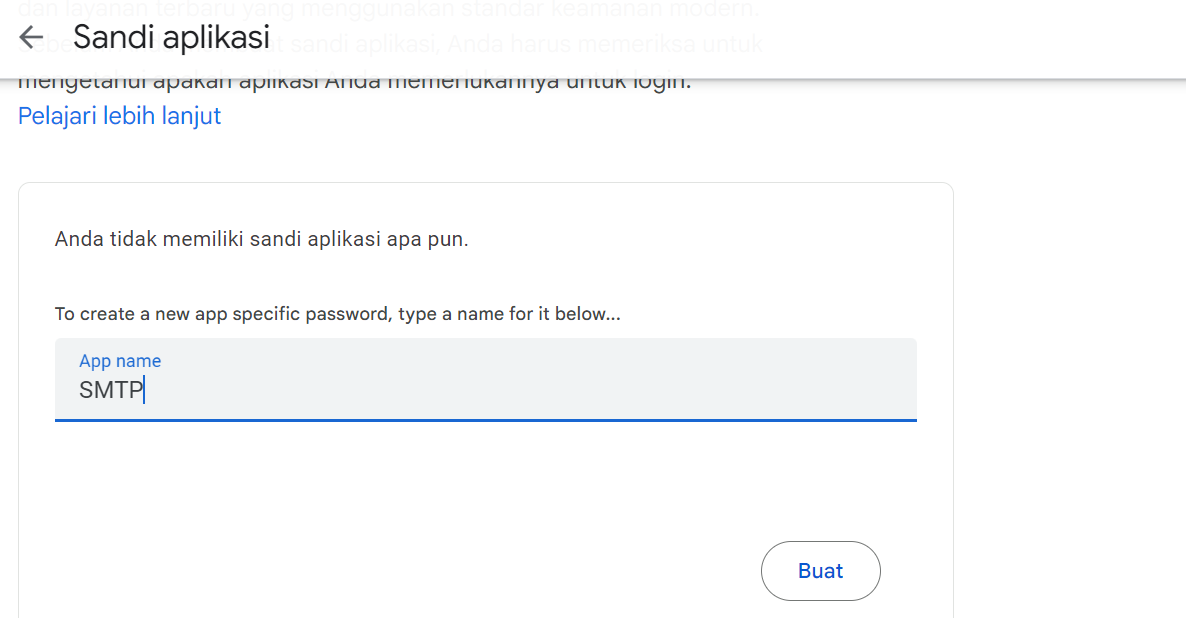


* + Gunakan layanan "gmail" dan berikan kredensial email dan kata sandi yang sesuai.

1. **Buat Variabel sendMail:**
   * Gunakan **transporter.sendMail** untuk mengirim email.
   * Tentukan properti dari, to, subject, dan text untuk mengisi informasi email.



1. **Mendapatkan Sandi Aplikasi:**
   * Untuk sandi, buka profil Google Chrome Anda dan pergi ke "Kelola Akun Google".
   * Pilih "Keamanan" dan temukan "Kata sandi aplikasi". Pilih aplikasi email dan perangkat yang sesuai, lalu buat sandi aplikasi.



1. **Gunakan Sandi Aplikasi:**
   * Gunakan sandi aplikasi yang telah dibuat sebagai kata sandi di dalam kode Anda.



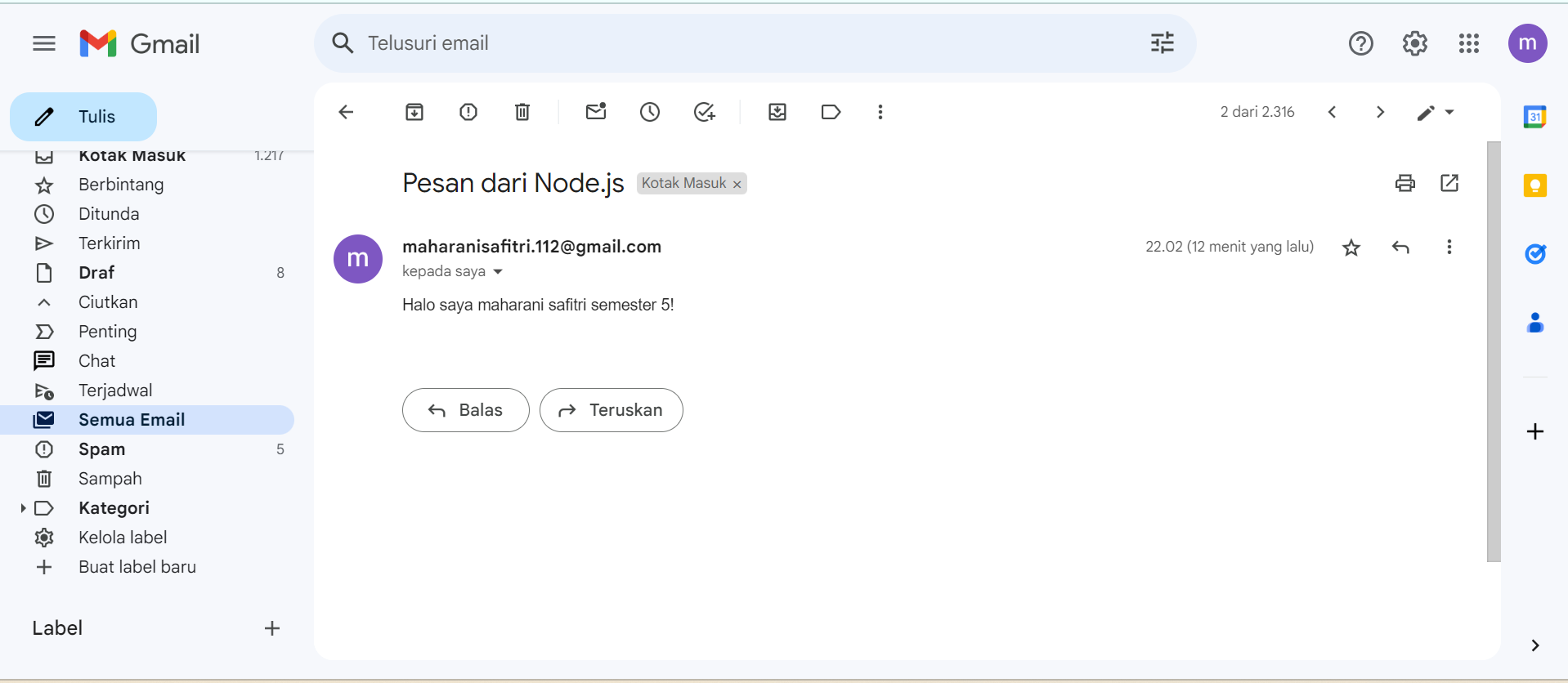
1. **Jalankan Fungsi dan Periksa Hasil:**
   * Panggil fungsi **kirimEmail** di bawahnya.



* + Simpan dan jalankan skrip Node.js dengan perintah **node nama\_app.js**.



1. **Periksa Hasil di Email Penerima:**
   * Setelah menjalankan skrip, buka email penerima untuk memeriksa pesan yang telah dikirim.
2. **Verifikasi Subjek dan Isi Email:**
   * Verifikasi bahwa subjek dan isi pesan email sesuai dengan yang telah Anda atur dalam kode.



C. DIAGARAM

Buatlah diagram beserta penjelasan terkait cara atau proses kerja masing-masing topik berikut IMAP, POP3, SMTP dan OAuth 2.0!

1. IMAP

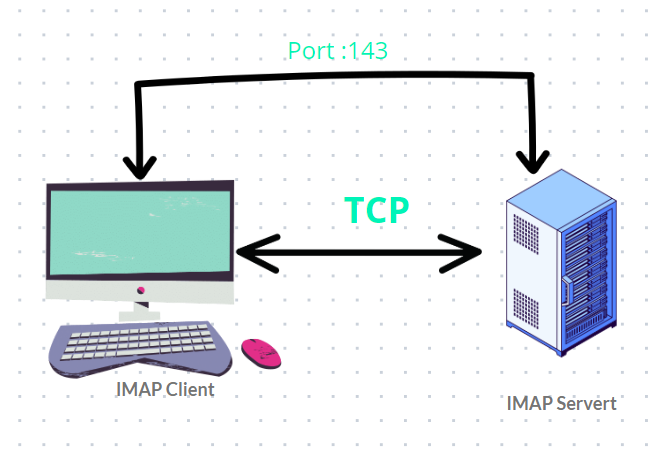


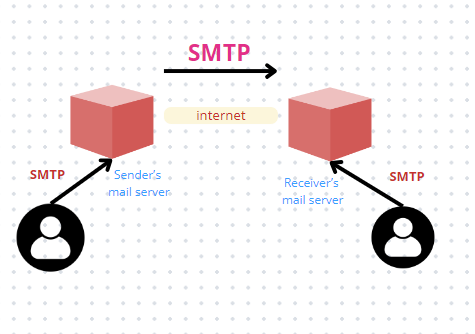
Diagram IMAP

IMAP (Internet Message Access Protocol) adalah protokol akses pesan standar yang memungkinkan pengguna untuk mengakses pesan email dari server email. IMAP memungkinkan pengguna untuk mengunduh pesan individu, folder, atau seluruh kotak surat. IMAP juga memungkinkan pengguna untuk mengedit pesan di server, yang berarti bahwa perubahan yang dilakukan pada pesan akan tercermin di semua perangkat yang mengakses pesan tersebut.

Proses kerja IMAP:

* Pengguna mengonfigurasi klien email mereka untuk menggunakan IMAP.
* Klien email terhubung ke server email menggunakan protokol IMAP.
* Server email memberikan daftar folder kepada klien email.
* Klien email memilih folder yang ingin diakses.
* Server email mengirimkan pesan-pesan dalam folder tersebut ke klien email.
* Klien email menampilkan pesan-pesan tersebut kepada pengguna.

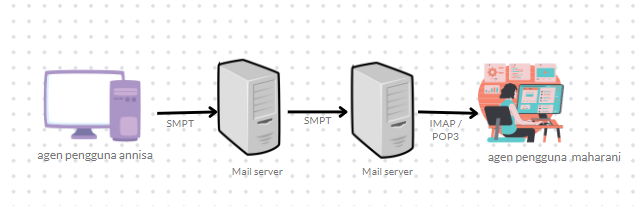
1. POP3



POP3 (Post Office Protocol 3) adalah protokol standar untuk mengunduh pesan email dari server email. POP3 memungkinkan pengguna untuk mengunduh seluruh kotak surat dari server ke perangkat lokal. Setelah pesan diunduh, pesan tersebut dihapus dari server.

Proses kerja POP3:

1. Pengguna mengonfigurasi klien email mereka untuk menggunakan POP3.
2. Klien email terhubung ke server email menggunakan protokol POP3.
3. Server email memberikan daftar folder kepada klien email.
4. Klien email memilih folder yang ingin diakses.
5. Server email mengirimkan seluruh pesan dalam folder tersebut ke klien email.
6. Klien email menyimpan pesan-pesan tersebut di perangkat lokal.
7. SMTP



Penjelasan:

SMTP (Simple Mail Transfer Protocol) adalah protokol standar untuk mengirim dan menerima email. SMTP digunakan oleh klien email untuk mengirim email ke server email, dan juga digunakan oleh server email untuk mengirim email ke server email lain.

Proses pengiriman email dimulai dari agen pengguna Annisa. Annisa ingin mengirim email ke Maharani. Annisa menggunakan program email client untuk membuat email dan memasukkan alamat email Maharani. Program email client kemudian akan menghubungkan Annisa ke server surat pengirim.

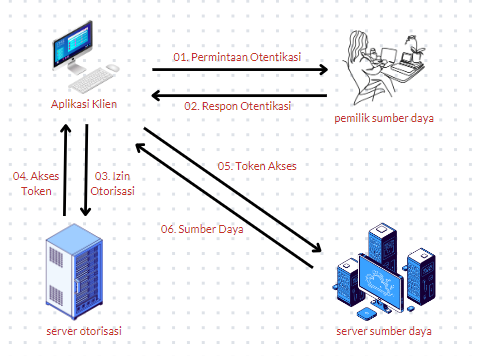
Server surat pengirim akan menerima email dari Annisa. Server surat pengirim akan menambahkan informasi tambahan ke email, seperti alamat IP server surat penerima. Kemudian, server surat pengirim akan mengirimkan email ke server surat penerima.

Server surat penerima akan menerima email dari server surat pengirim. Server surat penerima akan memeriksa alamat IP pengirim. Jika alamat IP pengirim valid, maka server surat penerima akan meneruskan email ke agen pengguna Maharani.

Agen pengguna Maharani akan menerima email dari server surat penerima. Agen pengguna Maharani akan menampilkan email kepada Maharani.

Berikut adalah penjelasan lebih rinci dari diagram tersebut:

* Agen pengguna Annisa adalah program yang digunakan oleh Annisa untuk membuat dan mengirim email.
* Server surat pengirim adalah server yang bertanggung jawab untuk mengirimkan email dari pengirim ke penerima.
* Server surat penerima adalah server yang bertanggung jawab untuk menerima email dari pengirim dan meneruskannya ke agen pengguna penerima.
* SMTP (Simple Mail Transfer Protocol) adalah protokol yang digunakan untuk mengirimkan email.
* Diagram tersebut menunjukkan bahwa proses pengiriman email melibatkan dua server surat dan dua agen pengguna. Proses pengiriman email dimulai dari agen pengguna pengirim dan berakhir di agen pengguna penerima.



Penjelasan:

OAuth 2.0 adalah protokol otorisasi yang memungkinkan aplikasi pihak ketiga untuk mengakses sumber daya di aplikasi lain tanpa perlu membagikan kredensial pengguna. OAuth 2.0 menggunakan konsep "token akses" untuk mewakili otorisasi pengguna untuk mengakses sumber daya.

Diagram menunjukkan langkah-langkah berikut:

1. Permintaan Otentikasi: Aplikasi klien mengirimkan permintaan otentikasi ke server otorisasi. Permintaan ini berisi informasi yang diperlukan untuk memverifikasi identitas pengguna, seperti nama pengguna dan kata sandi.
2. Respon Otentikasi: Server otorisasi memeriksa informasi dalam permintaan otentikasi. Jika informasi tersebut valid, server otorisasi akan mengirimkan respons otentikasi yang berisi token akses.
3. Izin: Token akses berisi informasi tentang izin yang dimiliki pengguna. Izin ini menentukan apa yang dapat dilakukan pengguna pada sumber daya.
4. Otorisasi: Server sumber daya menerima token akses dari aplikasi klien. Server sumber daya menggunakan informasi dalam token akses untuk menentukan apakah pengguna memiliki izin untuk mengakses sumber daya.
5. Akses: Jika pengguna memiliki izin untuk mengakses sumber daya, server sumber daya akan memberikan akses ke sumber daya.
6. Sumber Daya: Sumber daya adalah data atau layanan yang dapat diakses oleh pengguna.

Dalam konteks gambar, aplikasi klien adalah aplikasi yang digunakan pengguna untuk mengakses sumber daya. Server otorisasi adalah server yang bertanggung jawab untuk memverifikasi identitas pengguna. Server sumber daya adalah server yang berisi sumber daya yang ingin diakses pengguna.

1. Referensi:
2. <https://www.youtube.com/watch?v=0OYTeObVClU&t=32s>
3. <https://www.geeksforgeeks.org/internet-message-access-protocol-imap/>
4. <https://www.javatpoint.com/pop-protocol>
5. <https://images.app.goo.gl/q96iGGgBQsBBdV2q7>
6. <https://images.app.goo.gl/qb5n51XdjGzRT4g16>
7. **Kesimpulan**

IMAP, POP3, SMTP, dan OAuth 2.0 adalah protokol penting yang digunakan dalam komunikasi email. IMAP memungkinkan pengguna untuk mengakses pesan email dari server email, POP3 memungkinkan pengguna untuk mengunduh pesan email dari server email, SMTP digunakan untuk mengirim dan menerima email, dan OAuth 2.0 memungkinkan aplikasi pihak ketiga untuk mengakses sumber daya di aplikasi lain.

<https://www.geeksforgeeks.org/internet-message-access-protocol-imap/>