

GABRIELE ROMANATO

Menu

Come creare un sistema di logging delle visite di un sito web con Node.js

Registrazione delle visite di un sito web è essenziale per capire il comportamento degli utenti, ottimizzare le prestazioni e monitorare la sicurezza. In questo articolo, vedremo come creare un semplice sistema di logging delle visite utilizzando Node.js. Questo sistema sarà in grado di registrare i dati di accesso degli utenti in un file di log, inclusi dati come l'IP dell'utente, il timestamp della visita, l'URL visitato e altre informazioni opzionali.

Installa i moduli necessari: `express` per creare il server web e `morgan` per gestire il logging delle richieste.

```
npm install express morgan
```

Inizia creando un file chiamato `app.js` per configurare il server con Express. In questo file, imposteremo un semplice server web che risponde alle richieste dell'utente.

```
// app.js
const express = require('express');
const morgan = require('morgan');
const fs = require('fs');
const path = require('path');

const app = express();
const port = 3000;

// Crea un file di log in cui salvare le visite
const accessLogStream = fs.createWriteStream(path.join(__dirname, 'access.log'), {
  flags: 'a' });

// Configura morgan per loggare le richieste
app.use(morgan(':remote-addr - :method :url :status :response-time ms', { stream:
  accessLogStream }));

app.get('/', (req, res) => {
  return res.send('Benvenuto nel nostro sito!');
});

app.listen(port, () => {
  console.log(`Server in ascolto su http://localhost:${port}`);
});
```

Spiegazione del codice:

1. **Creazione del Server:** `app.listen()` avvia il server sulla porta 3000.
2. **Configurazione di Morgan:** morgan è configurato per scrivere i log nel file `access.log`. Il formato del log specificato include:
 - `:remote-addr`: l'indirizzo IP del visitatore.
 - `:method`: il metodo HTTP utilizzato (GET, POST, ecc.).
 - `:url`: l'URL richiesto.
 - `:status`: il codice di stato della risposta.
 - `:response-time`: il tempo impiegato dal server per rispondere.

Ogni visita verrà registrata con queste informazioni, che ci aiutano a comprendere le attività sul sito.

È possibile personalizzare i log di Morgan per includere ulteriori dettagli, come il `user-agent` o il `referrer`. Per fare ciò, basta modificare la stringa di configurazione di morgan.

```
app.use(morgan(':remote-addr - :method :url :status :response-time ms - :user-agent', { stream: accessLogStream }));
```

Conclusione

Abbiamo visto come creare un sistema di logging delle visite per un sito web con Node.js. Questo sistema raccoglie dati utili che possono migliorare l'esperienza dell'utente, garantire la sicurezza e ottimizzare le prestazioni del sito. Una configurazione più avanzata di logging potrebbe includere l'uso di database e strumenti di visualizzazione, ma questo sistema di base è già un ottimo punto di partenza per comprendere e monitorare l'attività sul tuo sito.

Applicazioni Correlate



-

Node.js Placeholder Image

Applicazione per la generazione con Node.js di immagini segnaposto.
Docker Docker Compose Node.js JavaScript ExpressJS



-

Node.js URL Shortener

Implementazione in Node.js di un sistema per l'abbreviazione degli URL.

Docker Docker Compose Node.js JavaScript Express JS MongoDB



-

JavaScript App Hash Change

Applicazione che sfrutta gli hash degli URL per gestire contenuto dinamico in JavaScript.
Docker Docker Compose Node.js JavaScript Express JS MySQL