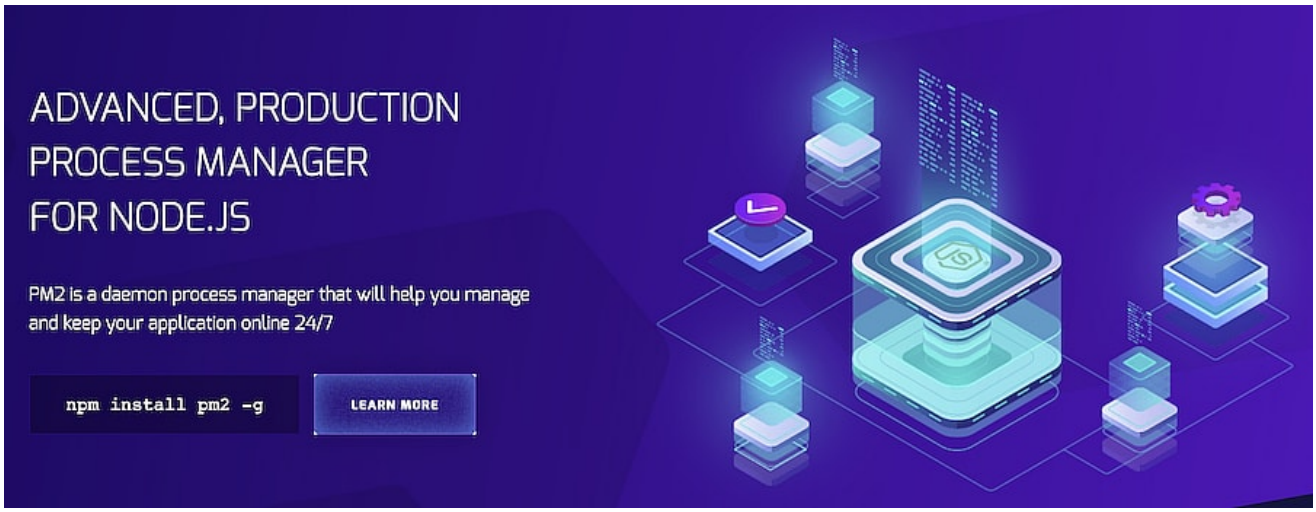
9 лучших инструментов мониторинга для NodeJS-приложений

Описание

Приходилось ли вам попадать в ситуацию, когда приложение не работает? Уведомил ли вас клиент о том, что услуги не работают? Наверное, это непростая ситуация. За прошедшие годы NodeJS стал популярным JavaScript-фреймворком для быстрой разработки микросервисов и API. При наличии рабочих нагрузок, работающих на NodeJS, крайне важно обеспечить правильный мониторинг основных показателей. Давайте рассмотрим следующие инструменты мониторинга Node.JS, которые помогут вам отслеживать производительность приложения и сервера, а также другие показатели.

Лучшие средства мониторинга приложений NodeJS

PM2



PM2 – самый известный инструмент, используемый новичками и опытными разработчиками для обеспечения постоянной работы своих приложений. PM2 используется для запуска рабочих нагрузок в реальном времени и их мониторинга из CLI или веб-интерфейса. Вы можете установить его как модуль NPM. PM2 – это готовое решение для предприятий, которое предоставляет следующие возможности из коробки.

- Управление несколькими NodeJS-приложениями на одном сервере
- Управление журналами приложений и ошибок
- Интеграция веб-интерфейса для мониторинга состояния и показателей приложения
- Автокластеризация для приложений NodeJS
- Горячая перезагрузка приложения.
- Поддержка сценариев запуска
- Вести потоковую передачу в веб-интерфейсе

Начало работы с PM2

Чтобы установить PM2, убедитесь, что у вас установлен NPM. Чтобы проверить установку и версию NPM, выполните следующую команду.

```
npm --version
```

Эта команда должна показать вам версию NPM, если она установлена. Далее установите PM2 с помощью следующей команды:

```
npm install -g pm2
```

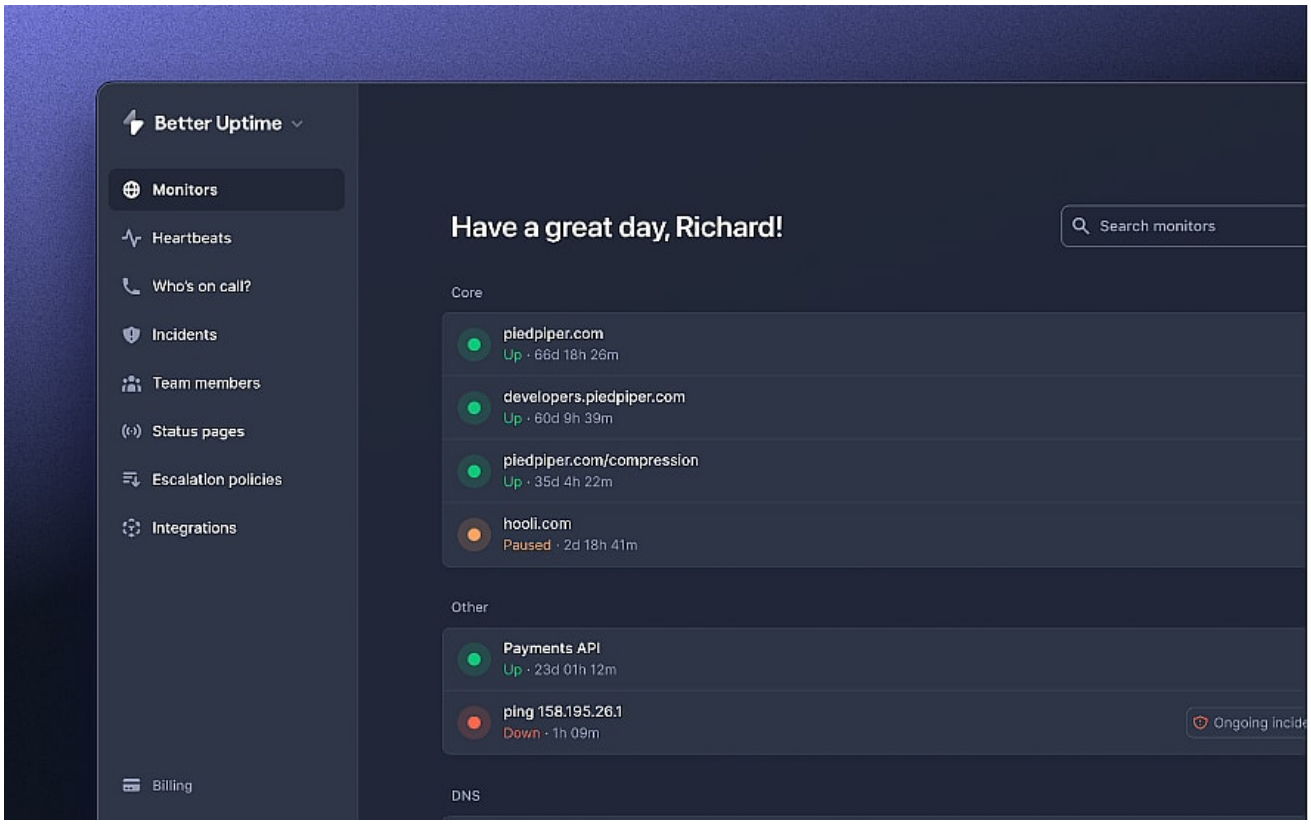
Глобальный флаг `-g` установит модуль и сделает его доступным глобально в виде команды. Чтобы запустить любое приложение с помощью PM2, перейдите в папку с проектом NodeJS и выполните приведенную ниже команду:

```
pm2 start <start-filename> --name myapp
```

Приведенная выше команда запускает приложение NodeJs, используя файл, указанный в первом аргументе. Она также присваивает приложению имя `myapp`. При выполнении NodeJS-приложения с помощью `pm2` он убеждается, что служба не упадет, а если упадет, то пытается автоматически перезапустить приложение. Чтобы включить мониторинг в реальном времени через веб-интерфейс, вам необходимо зарегистрироваться здесь. Как только ваши ключи будут сгенерированы, вы сможете увидеть инструкции по подключению PM2 на вашем сервере к веб-интерфейсу.

Better Uptime

Better Uptime – это современный сервис мониторинга, который объединяет мониторинг времени работы, управление инцидентами и страницы состояния в единый красиво оформленный продукт.



Настройка занимает 3 минуты. После этого вы получаете звонок, электронное письмо или уведомление из Slack, когда ваше приложение Node.js работает некорректно или полностью прекращает работу. Основные возможности:

- HTTP (s), Ping, API, истечение срока действия SSL и TLD, проверка заданий Cron
- Неограниченное количество оповещений о телефонных звонках
- Удобное планирование вызовов
- Скриншоты и журналы ошибок инцидентов
- Slack, Teams, Heroku, AWS и 100+ других интеграций.

Sematext

Sematext предлагает мониторинг всего стека. Полный стек означает, что у вас есть полная видимость доступности и производительности вашей инфраструктуры и приложений, метрики, журналы, пользовательский опыт фронтенда, здоровье и производительность ваших API, сертификаты SSL и т. д.

Sematext имеет более 100 интеграций, включая NodeJS. Вот некоторые из них:

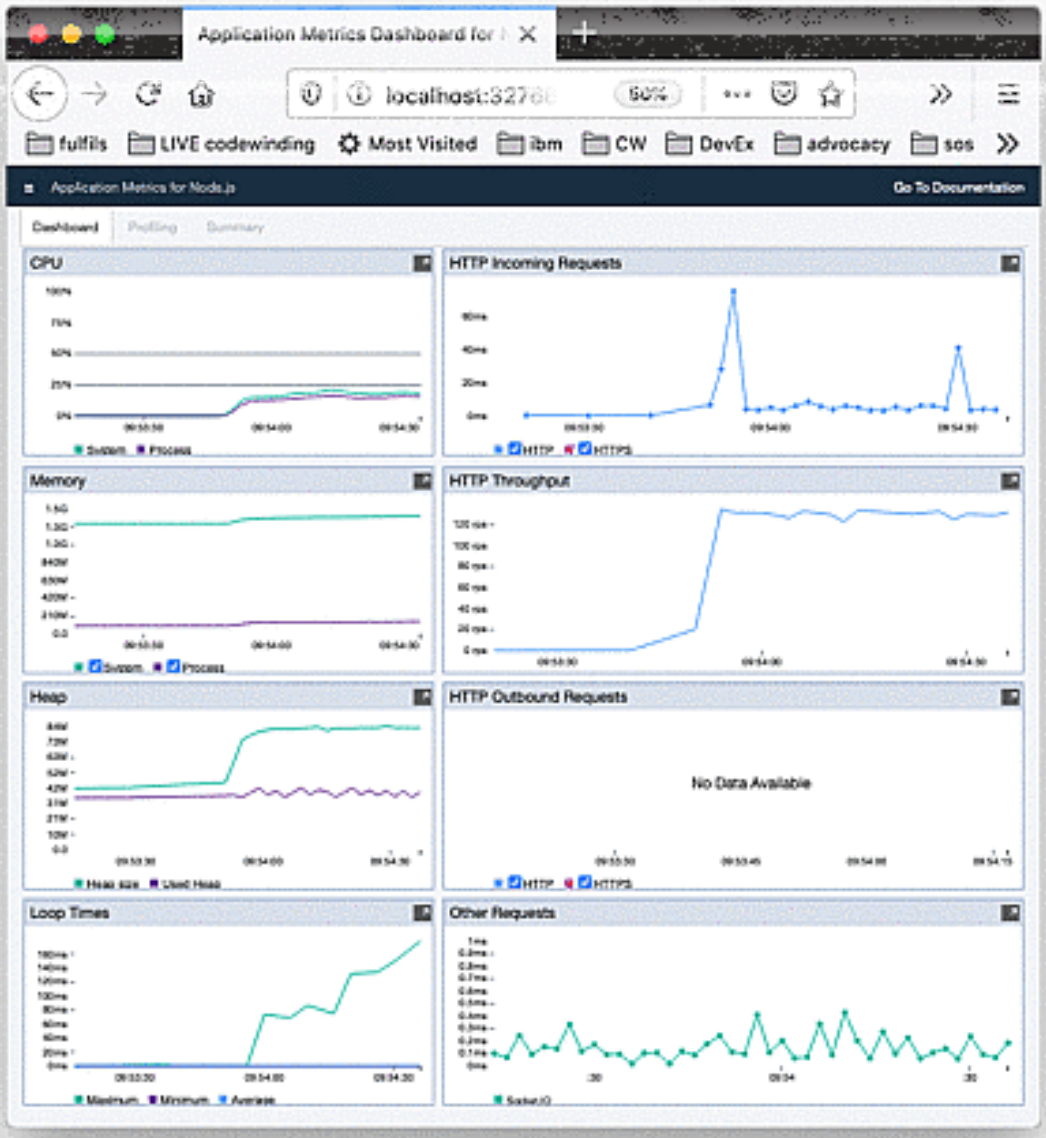
- Веб-серверы – Nginx, Apache
- Базы данных – MySQL, MariaDB, PostgreSQL, Redis
- Контейнеры – Kubernetes, Docker, Rancher
- Другие языки – PHP, Scala, Python, Go, Java
- AWS – EC2, ELB, S3, Aurora

И многое другое... Мониторинг NodeJS помогает отслеживать задержки, частоту запросов, сборку мусора, аномалии, рабочих, события, сердцебиение и многое другое. Вы можете визуализировать данные на красивой готовой панели или создать собственную при необходимости. Получайте уведомления, если что-то не в порядке через Slack, WebHooks, PageDuty, MS Teams и т. д. Sematext предлагает комплексное решение для мониторинга и анализа журналов “все в одном”. Вы можете проверить демо-версию, чтобы испытать платформу.

Appmetrics

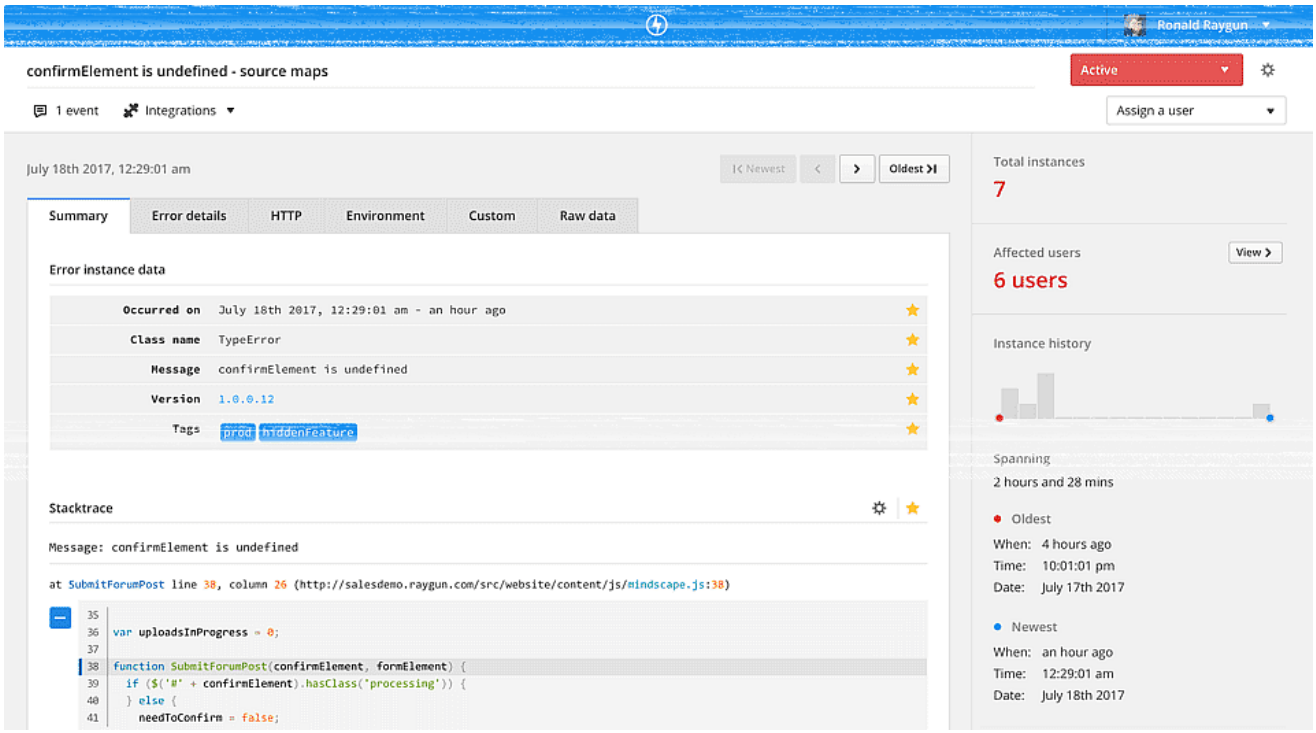
Appmetrics – это проект с открытым исходным кодом, управляемый и поддерживаемый компанией IBM. Проект нацелен на создание основы для сбора превосходных метрик приложения, охватывающих различные виды деятельности, такие как транзакция данных, скорость сети, производительность запросов к базе данных, использование процессора и памяти, сборка мусора и другие. Если

объединить его с таким плагином, как appmetrics-dash, можно получить отличный веб-интерфейс для мониторинга приложений.



Вы можете использовать Appmetrics в качестве промежуточного ПО для создания своих приложений мониторинга.

Raygun



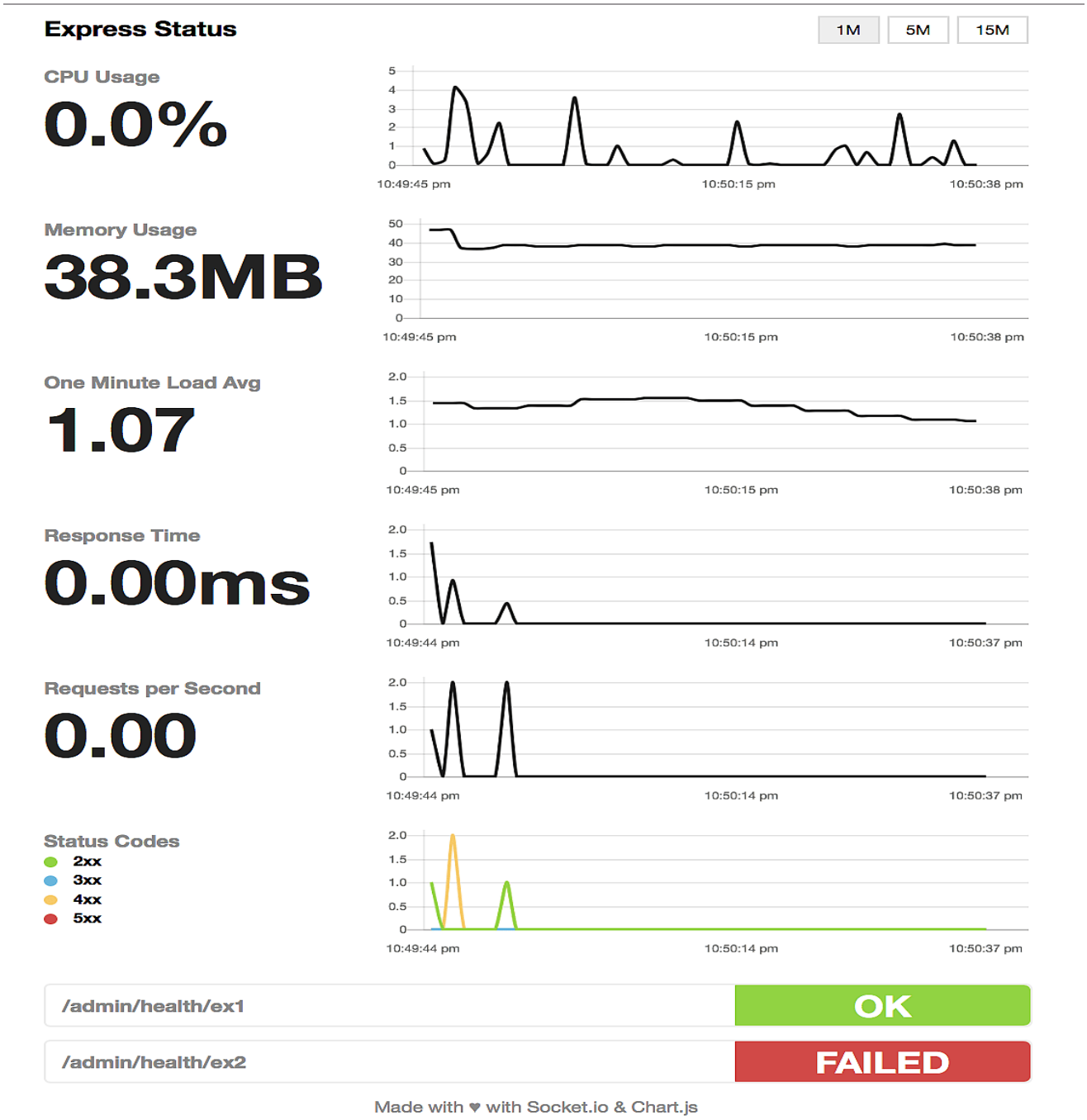
Raygun – это платформа для управления и мониторинга производительности приложений для нескольких языков программирования, включая NodeJS. Она помогает отлаживать проблемы в реальном времени с помощью журналов аварий, анализировать код приложения и находить нужную точку ошибки из самого кода. Некоторые из них перечислены ниже.

- Предоставляет оперативные и точные отчеты об ошибках
- Поддержка полностековых отчетов об ошибках
- Мониторинг сеансов пользователей в режиме реального времени
- Мониторинг производительности сервера и обнаружение “узких мест” в режиме реального времени

Raygun стоит того, чтобы потратить время и деньги на критически важные приложения. Он обеспечивает понимание, которое может привести к значительной окупаемости инвестиций.

Express Status Monitoring

Express Status Monitoring – это инструмент с открытым исходным кодом, предназначенный для мониторинга ExpressJS. Известно, что ExpressJS – самый популярный веб-фреймворк, используемый в NodeJS.



Как видно на приведенном выше снимке, вы можете отслеживать время отклика, частоту запросов, код состояния, загрузку процессора/памяти и среднюю нагрузку.

ClinicJS

Clinic.js – это библиотека, доступная для интеграции в NodeJS-приложения с целью мониторинга и получения критических параметров производительности. Она представляет собой комбинацию трех основных инструментов – **Doctor**, **Bubbleprof** и **Flame**. Каждый из этих инструментов используется для мониторинга различных метрик.

ClinicJS Доктор

- Сбор показателей здоровья и эвристики
- Автоматически вводите зонды в приложение
- Предоставьте графический интерфейс для просмотра производительности приложения и показателей в реальном времени
- Выработка рекомендаций по оптимизации приложений

ClinicJS Bubbleprof

- Стройте пузырьковые графики, описывающие внутреннее выполнение приложения
- Помогает выявить ошибки, показывая графики выполнения функций.
- Сбор метрик с помощью асинхронных крючков

- Предоставляет информацию о задержке между вызовами функций

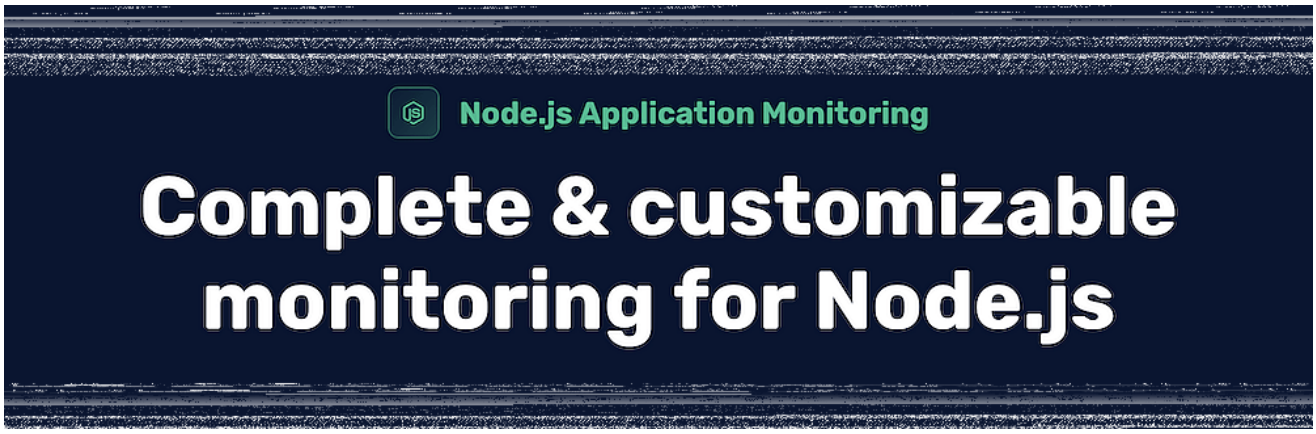
ClinicJS Flame

- Специальный мониторинг показателей процессора
- Графический пользовательский интерфейс для мониторинга производительности процессора приложения в реальном времени
- Генерирует графики пламени, помогающие определить максимальные значения, достигнутые метриками
- Предоставляет информацию о деятельности вашего приложения, потребляющей процессор.

Если вам нужна коммерческая поддержка, вы можете воспользоваться ею.

AppSignal

AppSignal – это коммерческий продукт, предлагающий автоинструментарий для HTTP-вызовов Node.js, Express, Next.js и Postgress.



Чтобы запустить его, нужно установить на сервер легкий агент AppSignal, который собирает необходимые метрики. В общих чертах он делает следующее.

- Мониторинг производительности – поиск и устранение ошибок в коде или инфраструктуре, замедляющих доступность и производительность приложения.
- Метрики – визуализируйте то, что для вас важно, на приборной панели.
- Обнаружение аномалий – получайте уведомления, когда все происходит не так, как вы ожидали.
- Monitor Server – не покупайте другую программу для мониторинга серверов. Вы

получаете все стандартные показатели ресурсов, такие как процессор, диск, память, сеть и т.д.

Prometheus + Grafana

Идея заключается в том, чтобы объединить два мощных программных обеспечения – Prometheus и Grafana. У Prometheus есть клиент для Nodejs под названием prom-client, установленный на сервере NodeJS для сбора метрик. Затем вы можете установить Grafana на том же сервере или удаленно, чтобы получать метрики из Prometheus для визуализации. Вы можете либо создать приборную панель Grafana с нуля, исходя из требований, либо воспользоваться двумя следующими.

- Приборная панель приложения NodeJS
- Метрики NodeJS

Для этого необходимо знать оба инструмента. Если вам нужна помощь в изучении Prometheus, ознакомьтесь с этим курсом на Udemy.

Заключение

Мониторинг часто игнорируется, но очень важен для обеспечения доступности приложений. Учитывая, что стоимость разработки является решающим фактором, инструменты, которые анализируют код, отслеживают производительность и дают

представление об ошибках, несомненно, имеют большое значение. Они экономят время и силы и помогают соблюдать SLA, быстрее предоставляя решение клиенту.

Дата Создания
17.02.2024