SSH Connection Timeout Fix

PS C:\Users\abdul> ssh -i "C:\Users\abdul\Downloads\taskManagement.pem" ubuntu@67.202.51.32 ssh: connect to host 67.202.51.32 port 22: Connection timed out when i try to connect ssh it gives me this error

my backend server cannot connect with db which is on mongodb atlas probably issue with ip what to do

now guide me i have made changes and push to the github i have deployed my frontend on the ec2 server so how to implement those change on the code which is on ec2 server

ubuntu@ip-10-0-1-52:~/task-management/frontend$ npm run build > taskmanagement@0.1.0 build > next build ▲ Next.js 15.5.2 Creating an optimized production build ... ✓ Compiled successfully in 5.7s Failed to compile. ./src/app/(auth)/login/page.tsx 90:16 Error: ' can be escaped with `&apos;`, `&lsquo;`, `&#39;`, `&rsquo;`. react/no-unescaped-entities ./src/app/dashboard/addTaskModal.tsx 9:26 Error: Unexpected any. Specify a different type. @typescript-eslint/no-explicit-any ./src/app/dashboard/editTaskModal.tsx 12:27 Error: Unexpected any. Specify a different type. @typescript-eslint/no-explicit-any 19:35 Error: React Hook "useState" is called conditionally. React Hooks must be called in the exact same order in every component render. Did you accidentally call a React Hook after an early return? react-hooks/rules-of-hooks 26:45 Error: React Hook "useState" is called conditionally. React Hooks must be called in the exact same order in every component render. Did you accidentally call a React Hook after an early return? react-hooks/rules-of-hooks 27:43 Error: React Hook "useState" is called conditionally. React Hooks must be called in the exact same order in every component render. Did you accidentally call a React Hook after an early return? react-hooks/rules-of-hooks 30:3 Error: React Hook "useEffect" is called conditionally. React Hooks must be called in the exact same order in every component render. Did you accidentally call a React Hook after an early return? react-hooks/rules-of-hooks 92:15 Error: Unexpected any. Specify a different type. @typescript-eslint/no-explicit-any ./src/app/dashboard/page.tsx 10:38 Error: Unexpected any. Specify a different type. @typescript-eslint/no-explicit-any 11:56 Error: Unexpected any. Specify a different type. @typescript-eslint/no-explicit-any 12:50 Error: Unexpected any. Specify a different type. @typescript-eslint/no-explicit-any 37:21 Error: Unexpected any. Specify a different type. @typescript-eslint/no-explicit-any 79:37 Error: Unexpected any. Specify a different type. @typescript-eslint/no-explicit-any 82:49 Error: Unexpected any. Specify a different type. @typescript-eslint/no-explicit-any 124:14 Warning: 'error' is defined but never used. @typescript-eslint/no-unused-vars info - Need to disable some ESLint rules? Learn more here: https://nextjs.org/docs/app/api-reference/config/eslint#disabling-rules

ubuntu@ip-10-0-1-52:~/task-management/frontend$ npm run build > taskmanagement@0.1.0 build > next build ▲ Next.js 15.5.2 Creating an optimized production build ... ✓ Compiled successfully in 3.3s Skipping validation of types Skipping linting ✓ Collecting page data Error occurred prerendering page "/dashboard". Read more: https://nextjs.org/docs/messages/prerender-error ReferenceError: localStorage is not defined at j (.next/server/app/dashboard/page.js:1:8325) { digest: '3422788247' } Export encountered an error on /dashboard/page: /dashboard, exiting the build. ✕ Next.js build worker exited with code: 1 and signal: null this was the response what to do next

"use client"; import React, { useState, useEffect } from "react"; import Cookies from "js-cookie"; import AddTaskForm from "./addTaskModal"; import EditTask from "./editTaskModal"; import { useRouter } from "next/navigation"; export default function ManagerDashboard() { const [tasks, setTasks] = useState<any[]>([]); const [selectedTaskId, setSelectedTaskId] = useState<any>(null); const [taskDetails, setTaskDetails] = useState<any>(null); const token = Cookies.get("auth_token"); const [editTaskModal, setEditTaskModal] = useState(false); const [addTaskModal, setAddTaskModal] = useState(false); const [loading, setLoading] = useState(true); const router = useRouter(); useEffect(() => { const fetchTasks = async () => { setLoading(true); try { const response = await fetch("http://localhost:3000/task", { method: "GET", headers: { "Content-Type": "application/json", Authorization: `Bearer ${token}`, }, }); if (!response.ok) { throw new Error("Failed to fetch tasks"); } const data = await response.json(); setTasks(data.Task); } catch (err: any) { console.error(err); } finally { setLoading(false); } }; fetchTasks(); }, [token]); useEffect(() => { if (!selectedTaskId) return; const fetchTaskById = async () => { try { const response = await fetch(`http://localhost:3000/task/id/${selectedTaskId}`, { headers: { Authorization: `Bearer ${token}` }, }); if (!response.ok) throw new Error("Failed to fetch task details"); const data = await response.json(); setTaskDetails(data.Task); } catch (error) { console.error(error); } }; fetchTaskById(); }, [selectedTaskId, token]); const handleTaskClick = (id: string) => { setSelectedTaskId(id); }; const closeSidebar = () => { setSelectedTaskId(null); setTaskDetails(null); }; const handleTaskAdded = (newTask: any) => { setTasks((prev) => [...prev, newTask]); }; const handleTaskUpdated = async (updatedTask: any) => { setTasks((prev) => prev.map((task) => (task._id === updatedTask._id ? updatedTask : task))); if (selectedTaskId === updatedTask._id) { try { const response = await fetch(`http://localhost:3000/task/id/${selectedTaskId}`, { headers: {

Authorization: `Bearer ${token}` }, }); if (!response.ok) throw new Error("Failed to fetch task details"); const data = await response.json(); setTaskDetails(data.Task); } catch (error) { console.error(error); } } }; const handleLogout = async () => { Cookies.remove("auth_token"); router.push("/login"); }; const handleDeleteTask = async (id: string) => { try { const response = await fetch(`http://localhost:3000/task/${id}`, { method: "DELETE", headers: { "Content-Type": "application/json", Authorization: `Bearer ${token}`, }, }); if (!response.ok) { throw new Error("Failed to delete task"); } setTasks((prevTask) => prevTask.filter((task) => task._id !== id)); console.log("Task deleted successfully"); } catch (error) { console.error("Error Deleting task"); } }; const completedTasks = tasks.filter((t) => t.status === "completed"); const inProgressTasks = tasks.filter((t) => t.status === "in_progress"); const toDoTasks = tasks.filter((t) => t.status === "to-do"); const username = localStorage.getItem("UserName"); if (loading) { return (   ); } this is dashboard page.tsx

ubuntu@ip-10-0-1-52:~/task-management/frontend$ npm run build > taskmanagement@0.1.0 build > next build ▲ Next.js 15.5.2 Creating an optimized production build ... ✓ Compiled successfully in 5.1s Skipping validation of types Skipping linting ✓ Collecting page data ✓ Generating static pages (8/8) ✓ Collecting build traces ✓ Finalizing page optimization Route (app) Size First Load JS ┌ ○ / 123 B 102 kB ├ ○ /_not-found 994 B 103 kB ├ ○ /dashboard 3.99 kB 106 kB ├ ○ /login 1.73 kB 104 kB └ ○ /signup 1.88 kB 104 kB + First Load JS shared by all 102 kB ├ chunks/255-e3bf15caf1f1e0f9.js 45.7 kB ├ chunks/4bd1b696-c023c6e3521b1417.js 54.2 kB └ other shared chunks (total) 1.92 kB ƒ Middleware 34.1 kB ○ (Static) prerendered as static content is this mean build has successfully compiled

now i want to install nginx to connect my frontend with my backend as both are on same ec2 instance so where i install nginx ? folder structure is taskmanagement main folder inside 2 folder frontend and backend

where i run these command inside which folder

now guide me how to create and where to create config file for nginx and what to write in config file

my instance of ec2 does not have elastic ip address and public dns ? what to do for this

when go into action and then in netwroking there is no option for associate elastic ip rather there is option for disassociate elastic ip and when i click over it , it tell me The selected instance does not have an associated Elastic IP address.

now lets move forward on configuring nginx

here i want a little first guide me it is possible i want this so i don't change any thing in frontend and on backend all i want to do something woth configuration i want that in config for backend i define location with /api/ but when it tranfer to backend it remove api word from the request and just forward the request is this possible and how?

what is the difference between this response config file and previous ersponse config file both are same?

also tell mw why we have added this url when we are calling like this http://localhost:4000 in frontend const response = await fetch("http://localhost:3000/task", { method: "GET", headers: { "Content-Type": "application/json", Authorization: `Bearer ${token}`, }, });

i am confused that in nginx config in proxy_pass which i have to call http://127.0.0.1:3000; or http://localhost:3000?

how do i check on which port my backend is running

my backend is running on port 4000 and i want to run on port 3000 how to do it

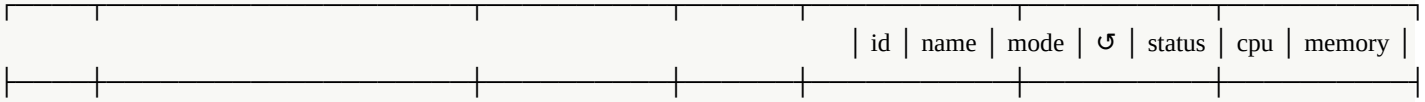
i have changed it but still it is runnig on port 4000 but there is no env also may be i think so it is getting from the build?am i correct

/home/ubuntu/task-management/backend/dist/main.js

ubuntu@ip-10-0-1-52:~/task-management/backend$ tsc Command 'tsc' not found, but can be installed with: sudo apt install node-typescript i have created build by using npm run build

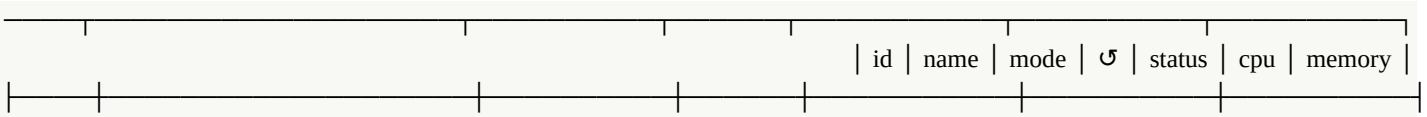
everything is done like frontend is runnig backend is running nginx is working and i have done nginx configuration still when i open elastic ip of instance it is giving me nginx page not my code page

ubuntu@ip-10-0-1-52:~/task-management$ pm2 ls



id	name	mode	↺	status	cpu	memory
0	backend	fork	0	online	0%	95.2mb
1	frontend	fork	15	errored	0%	0b

/home/ubuntu/.pm2/logs/frontend-error.log last 15 lines: 1|frontend | errno: -98, 1|frontend | syscall: 'listen', 1|frontend | address: '::', 1|frontend | port: 3000 1|frontend | } 1|frontend | ✕ Failed to start server 1|frontend | Error: listen EADDRINUSE: address already in use :::3000 1|frontend | at <unknown> (Error: listen EADDRINUSE: address already in use :::3000) 1|frontend | at new Promise (<anonymous>) { 1|frontend | code: 'EADDRINUSE', 1|frontend | errno: -98, 1|frontend | syscall: 'listen', 1|frontend | address: '::', 1|frontend | port: 3000 1|frontend | } if port is busy it automatically shift to port 3001



id	name	mode	↺	status	cpu	memory
0	backend	fork	0	online	0%	96.3mb
1	frontend	fork	15	errored	0%	0b
2	frontend	fork	0	online	0%	17.8mb

how to remove id 1 frontend

my frontend is displaying on the the ip but there is backend issue when i try to login it is not logging in page-61fcf9b045a25e59.js:1 POST http://localhost:3000/user/login net::ERR_CONNECTION_REFUSED l @ page-61fcf9b045a25e59.js:1 i8 @ 4bd1b696-c023c6e3521b1417.js:1 (anonymous) @ 4bd1b696-c023c6e3521b1417.js:1 nz @ 4bd1b696-c023c6e3521b1417.js:1 sn @ 4bd1b696-c023c6e3521b1417.js:1 cc @ 4bd1b696-c023c6e3521b1417.js:1 ci @ 4bd1b696-c023c6e3521b1417.js:1Understand this error page-61fcf9b045a25e59.js:1 TypeError: Failed to fetch at l (page-61fcf9b045a25e59.js:1:1877) at i8 (4bd1b696-c023c6e3521b1417.js:1:135363) at 4bd1b696-c023c6e3521b1417.js:1:141449 at nz (4bd1b696-c023c6e3521b1417.js:1:19197) at sn (4bd1b696-c023c6e3521b1417.js:1:136596) at cc (4bd1b696-c023c6e3521b1417.js:1:163598) at ci (4bd1b696-c023c6e3521b1417.js:1:163420) this is the error

but i have done this same process with little change browser request for backend with api when it forward to the backend it remove api word from the api request

how to open nginx vonfig file to update the configuration

page-61fcf9b045a25e59.js:1 POST http://localhost:3000/user/login net::ERR_CONNECTION_REFUSED l @ page-61fcf9b045a25e59.js:1 i8 @ 4bd1b696-c023c6e3521b1417.js:1 (anonymous) @ 4bd1b696-c023c6e3521b1417.js:1 nz @ 4bd1b696-c023c6e3521b1417.js:1 sn @ 4bd1b696-c023c6e3521b1417.js:1 cc @ 4bd1b696-c023c6e3521b1417.js:1 ci @ 4bd1b696-c023c6e3521b1417.js:1Understand this error page-61fcf9b045a25e59.js:1 TypeError: Failed to fetch at l (page-61fcf9b045a25e59.js:1:1877) at i8 (4bd1b696-c023c6e3521b1417.js:1:135363) at 4bd1b696-c023c6e3521b1417.js:1:141449 at nz (4bd1b696-c023c6e3521b1417.js:1:19197) at sn (4bd1b696-c023c6e3521b1417.js:1:136596) at cc (4bd1b696-c023c6e3521b1417.js:1:163598) at ci (4bd1b696-c023c6e3521b1417.js:1:163420) page-61fcf9b045a25e59.js:1 POST http://localhost:3000/user/login net::ERR_CONNECTION_REFUSED l @ page-61fcf9b045a25e59.js:1 i8 @ 4bd1b696-c023c6e3521b1417.js:1 (anonymous) @ 4bd1b696-c023c6e3521b1417.js:1 nz @ 4bd1b696-c023c6e3521b1417.js:1 sn @ 4bd1b696-c023c6e3521b1417.js:1 cc @ 4bd1b696-c023c6e3521b1417.js:1 ci @ 4bd1b696-c023c6e3521b1417.js:1Understand this error page-61fcf9b045a25e59.js:1 TypeError: Failed to fetch at l (page-61fcf9b045a25e59.js:1:1877) at i8 (4bd1b696-c023c6e3521b1417.js:1:135363) at 4bd1b696-c023c6e3521b1417.js:1:141449 at nz (4bd1b696-c023c6e3521b1417.js:1:19197) at sn

(4bd1b696-c023c6e3521b1417.js:1:136596) at cc (4bd1b696-c023c6e3521b1417.js:1:163598) at ci (4bd1b696-c023c6e3521b1417.js:1:163420) still not working what to do?

GNU nano 7.2 /etc/nginx/sites-available/taskmanagement server { listen 80; server_name _; # FRONTEND location / { proxy_pass http://127.0.0.1:3001; proxy_http_version 1.1; proxy_set_header Upgrade $http_upgrade; proxy_set_header Connection 'upgrade'; proxy_set_header Host $host; proxy_cache_bypass $http_upgrade; } # BACKEND (strip /api) location /api/ { rewrite ^/api/(.*)$ /$1 break; proxy_pass http://127.0.0.1:3000/; proxy_http_version 1.1; proxy_set_header Host $host; proxy_set_header X-Real-IP $remote_addr; proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for; } } ths is my config file of nginx is this correct

then why it is showing me this error page-9e757dfef65c4e05.js:1 POST http://localhost:3000/user/signup net::ERR_CONNECTION_REFUSED i @ page-9e757dfef65c4e05.js:1 i8 @ 4bd1b696-c023c6e3521b1417.js:1 (anonymous) @ 4bd1b696-c023c6e3521b1417.js:1 nz @ 4bd1b696-c023c6e3521b1417.js:1 sn @ 4bd1b696-c023c6e3521b1417.js:1 cc @ 4bd1b696-c023c6e3521b1417.js:1 ci @ 4bd1b696-c023c6e3521b1417.js:1Understand this error page-9e757dfef65c4e05.js:1 TypeError: Failed to fetch at i (page-9e757dfef65c4e05.js:1:1953) at i8 (4bd1b696-c023c6e3521b1417.js:1:135363) at 4bd1b696-c023c6e3521b1417.js:1:141449 at nz (4bd1b696-c023c6e3521b1417.js:1:19197) at sn (4bd1b696-c023c6e3521b1417.js:1:136596) at cc (4bd1b696-c023c6e3521b1417.js:1:163598) at ci (4bd1b696-c023c6e3521b1417.js:1:163420)

C:\Users\abdul>ssh -i "C:\Users\abdul\Downloads\taskManagement.pem" ubuntu@52.45.96.186 ssh: connect to host 52.45.96.186 port 22: Connection timed out i don't know why i can't ssh

const handleSubmit = async (e: React.FormEvent<HTMLFormElement>) => { e.preventDefault(); try { const response = await fetch("http://localhost:3000/user/login", { method: "POST", headers: { "Content-Type": "application/json" }, body: JSON.stringify(formData), }); if (!response.ok) { throw new Error("Login Failed"); } const data = await response.json(); Cookies.set("auth_token", data.Data.token, { expires: 1, sameSite: "lax", }); this is the code now tell me what i have to change like user/login or api/user/login? cause i have written this in nginx configuration # BACKEND API location /api/ { rewrite ^/api/(.*)$ /$1 break; # remove /api proxy_pass http://localhost:3000/; proxy_http_version 1.1; proxy_set_header Host $host; proxy_set_header X-Real-IP $remote_addr; proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for; }

page-61fcf9b045a25e59.js:1 POST http://localhost:3000/user/login net::ERR_CONNECTION_REFUSED l @ page-61fcf9b045a25e59.js:1 i8 @ 4bd1b696-c023c6e3521b1417.js:1 (anonymous) @ 4bd1b696-c023c6e3521b1417.js:1 nz @ 4bd1b696-c023c6e3521b1417.js:1 sn @ 4bd1b696-c023c6e3521b1417.js:1 cc @ 4bd1b696-c023c6e3521b1417.js:1 ci @ 4bd1b696-c023c6e3521b1417.js:1Understand this error page-61fcf9b045a25e59.js:1 TypeError: Failed to fetch at l (page-61fcf9b045a25e59.js:1:1877) at i8 (4bd1b696-c023c6e3521b1417.js:1:135363) at 4bd1b696-c023c6e3521b1417.js:1:141449 at nz (4bd1b696-c023c6e3521b1417.js:1:19197) at sn (4bd1b696-c023c6e3521b1417.js:1:136596) at cc (4bd1b696-c023c6e3521b1417.js:1:163598) at ci (4bd1b696-c023c6e3521b1417.js:1:163420)

const handleSubmit = async (e: React.FormEvent<HTMLFormElement>) => { e.preventDefault(); try { const response = await fetch("http://localhost:3000/user/login", { method: "POST", headers: { "Content-Type": "application/json" }, body: JSON.stringify(formData), }); if (!response.ok) { throw new Error("Login Failed"); } const data = await response.json(); Cookies.set("auth_token", data.Data.token, { expires: 1, sameSite: "lax", }); localStorage.setItem("UserName", data.Data.username); localStorage.setItem("Email", data.Data.email); router.push("/dashboard"); } catch (error) { console.log(error); } }; const handleSubmit = async (e: React.FormEvent<HTMLFormElement>) => { e.preventDefault(); try { const response = await fetch("http://localhost:3000/user/signup", { method: "POST", headers: { "Content-Type": "application/json" }, body: JSON.stringify(formData), }); if (!response.ok) { throw new Error("Signup Failed"); } const data = await response.json(); console.log("Response: " , data) localStorage.setItem("UserName", data.data.username); localStorage.setItem("Email", data.data.email); Cookies.set("auth_token", data.data.token, { expires: 1, sameSite: "lax", }); router.push("/dashboard"); } catch (error) { console.log(error); } }; try { const response = await fetch("http://localhost:3000/task", { method: "POST", headers: { "Content-Type": "application/json", Authorization: `Bearer ${token}`, }, body: JSON.stringify(payload), }); if (!response.ok) { throw new Error("Failed to add task"); } const data = await response.json(); onTaskAdded(data.Task) console.log("Task Added Successfully"); onClose(); } catch (error) { console.error(error); } }; try { const response = await fetch(`http://localhost:3000/task/${taskId}`, { method: "PATCH", headers: { "Content-Type": "application/json", Authorization: `Bearer ${token}`, }, body: JSON.stringify(body), }); const data = await response.json(); if (!response.ok) { throw new Error(data.message || "Failed to update task"); } const updatedTask = data.Task; onTaskUpdate(updatedTask); onClose(); } catch (error) { console.error(error); } }; useEffect(() => { const fetchTask = async () => { try {

const response = await fetch(`http://localhost:3000/task/id/${taskId}`, { method: "GET", headers: { Authorization: `Bearer ${token}` }, }); if (!response.ok) throw new Error("Failed to fetch task"); const data = await response.json(); const task = data.Task; const formatDate = (dateString: string) => { if (!dateString) return ""; const date = new Date(dateString); return date.toISOString().split("T")[0]; }; useEffect(() => { const fetchTasks = async () => { setLoading(true); try { const response = await fetch("http://localhost:3000/task", { method: "GET", headers: { "Content-Type": "application/json", Authorization: `Bearer ${token}`, }, }); const fetchTaskById = async () => { try { const response = await fetch(`http://localhost:3000/task/id/${selectedTaskId}`, { headers: { Authorization: `Bearer ${token}` }, }); const handleTaskUpdated = async (updatedTask: any) => { setTasks((prev) => prev.map((task) => (task._id === updatedTask._id ? updatedTask : task))); if (selectedTaskId === updatedTask._id) { try { const response = await fetch(`http://localhost:3000/task/id/${selectedTaskId}`, { headers: { Authorization: `Bearer ${token}` }, }); if (!response.ok) throw new Error("Failed to fetch task details"); const handleDeleteTask = async (id: string) => { try { const response = await fetch(`http://localhost:3000/task/${id}`, { method: "DELETE", headers: { "Content-Type": "application/json", Authorization: `Bearer ${token}`, }, }); now tell me what to make changes in the api call in frontend these are the frontend code

ubuntu@ip-10-0-1-52:~/task-management/frontend$ git pull origin main Username for 'https://github.com': Abdullah-Qureshi-404 Password for 'https://Abdullah-Qureshi-404@github.com': remote: Enumerating objects: 25, done. remote: Counting objects: 100% (25/25), done. remote: Compressing objects: 100% (5/5), done. remote: Total 13 (delta 6), reused 13 (delta 6), pack-reused 0 (from 0) Unpacking objects: 100% (13/13), 1.07 KiB | 136.00 KiB/s, done. From https://github.com/Abdullah-Qureshi-404/Task-Management-Frontend- * branch main -> FETCH_HEAD 21608d2..011cb25 main -> origin/main Updating 21608d2..011cb25 error: Your local changes to the following files would be overwritten by merge: src/app/(auth)/login/page.tsx Please commit your changes or stash them before you merge. Aborting is there any issue?

not just login i have changed other many files also? and i have pushed to github now i want to update my code on ec2 instand frontend

now guide me step by step to apply application load balancer and auto scalling to my project i have completed my deployment of frontend and backend on ec2 server and they both are connected and working properly

there is an issue that my backend is runnig on port 3000 when frontend run it gives error so i manually run my frontend on port 3001 PORT=3001 pm2 start npm --name "frontend" -- start through this ccommand so when pm2 autostart will it automaticallly start the frontend on port 3001 or this error will happen there also

ubuntu@ip-10-0-1-52:~/task-management/frontend$ pm2 ls

id	name	mode	↺	status	cpu	memory
0	backend	fork	0	online	0%	99.0mb
4	frontend	fork	0	online	0%	54.5mb

ubuntu@ip-10-0-1-52:~/task-management/frontend$ ss -tulnp | grep node tcp LISTEN 0 511 *:3000 *:* users:(("node /home/ubun",pid=1377,fd=23))

ubuntu@ip-10-0-1-52:~/task-management/frontend$ pm2 describe frontend Describing process with id 4 - name frontend

| status | online | | name | frontend | | namespace | default | | version | N/A | | restarts | 0 | | uptime | 2m | | script path | /usr/bin/npm | | script args | start | | error log path | /home/ubuntu/.pm2/logs/frontend-error.log | | out log path | /home/ubuntu/.pm2/logs/frontend-out.log | | pid path | /home/ubuntu/.pm2/pids/frontend-4.pid | | interpreter | /usr/bin/node | | interpreter args | N/A | | script id | 4 | | exec cwd | /home/ubuntu/task-management/frontend | | exec mode | fork_mode | | node.js version | 20.19.6 | | node env | production | | watch & reload | ✘ | | unstable restarts | 0 | | created at | 2025-12-25T10:27:08.871Z |

Actions available

| km:heapdump | | km:cpu:profiling:start | | km:cpu:profiling:stop | | km:heap:sampling:start | | km:heap:sampling:stop |

Trigger via: pm2 trigger frontend <action_name>

Code metrics value

| Used Heap Size | 7.42 MiB | | Heap Usage | 86.31 % | | Heap Size | 8.60 MiB | | Event Loop Latency p95 | 1.24 ms | | Event Loop Latency | 0.37 ms | | Active handles | 5 | | Active requests | 0 |

Divergent env variables

from local env Add your own code metrics: http://bit.ly/code-metrics Use `pm2 logs frontend [--lines 1000]` to display logs Use `pm2 env 4` to display environment variables Use `pm2 monit` to monitor CPU and Memory usage frontend

ubuntu@ip-10-0-1-52:~/task-management/frontend$ pm2 describe frontend Describing process with id 5 - name frontend
┌ status │ online │ │ name │ frontend │ │ namespace │ default │ │ version │ N/A │ │ restarts │ 0 │ │ uptime │ 15s │ │ script path │ /usr/bin/npm │ │ script args │ start │ │ error log path │ /home/ubuntu/.pm2/logs/frontend-error.log │ │ out log path │ /home/ubuntu/.pm2/logs/frontend-out.log │ │ pid path │ /home/ubuntu/.pm2/pids/frontend-5.pid │ │ interpreter │ /usr/bin/node │ │ interpreter args │ N/A │ │ script id │ 5 │ │ exec cwd │ /home/ubuntu/task-management/frontend │ │ exec mode │ fork_mode │ │ node.js version │ 20.19.6 │ │ node env │ production │ │ watch & reload │ ✘ │ │ unstable restarts │ 0 │ │ created at │ 2025-12-25T10:32:22.807Z │
└ Actions available
┌ km:heapdump │ │ km:cpu:profiling:start │ │ km:cpu:profiling:stop │ │ km:heap:sampling:start │ │ km:heap:sampling:stop │
└ Trigger via: pm2 trigger frontend <action_name> Code metrics value
┌ Used Heap Size │ 7.28 MiB │ │ Heap Usage │ 87.16 % │ │ Heap Size │ 8.35 MiB │ │ Event Loop Latency p95 │ 0.87 ms │ │ Event Loop Latency │ 0.01 ms │ │ Active handles │ 5 │ │ Active requests │ 0 │
└ Divergent env variables from local env Add your own code metrics: http://bit.ly/code-metrics Use `pm2 logs frontend [--lines 1000]` to display logs Use `pm2 env 5` to display environment variables Use `pm2 monit` to monitor CPU and Memory usage frontend ubuntu@ip-10-0-1-52:~/task-management/frontend$ ss -tulnp | grep node tcp LISTEN 0 511 *:3000 *:* users:(("node /home/ubun",pid=1377,fd=23))

once i ran my both frontend and backend and if they creashes so will pm2 automatically run the backend on port 3000 and frontend on 3001?

503 Service Temporarily Unavailable when i paste my dns of load balancer on browser it show me thia

ubuntu@ip-10-0-1-52:~$ pm2 ls

id	name	mode	↺	status	cpu	memory
0	backend	fork	0	online	0%	98.1mb
1	frontend	fork	0	online	0%	55.1mb

ubuntu@ip-10-0-1-52:~$ ss -tulnp | grep node tcp LISTEN 0 511 *:3000 *:* users:(("node /home/ubun",pid=850,fd=23)) Target group: task-management Details Targets Monitoring Health checks Attributes Tags Details arn:aws:elasticloadbalancing:us-east-1:590183872712:targetgroup/task-management/a57bcf96b9a5fab2 Target type Instance Protocol : Port HTTP: 80 Protocol version HTTP1 VPC vpc-0ac96cefdfa19b9f1 IP address type IPv4 Load balancer None associated 0 Total targets 0 Healthy 0 Anomalous 0 Unhealthy 0 Unused 0 Initial 0 Draining Target group: task-management Details Targets Monitoring Health checks Attributes Tags Registered targets (0) Info Anomaly mitigation: Not applicable Deregister Register targets Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets. 1 Instance ID Name Port Zone Health status Health status details Administrative override Override details Launch time No registered targets You have not registered targets to this group yet Register targets

Create launch template Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions. Launch template name and description Launch template name - required task-management-template Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'. Template version description Max 255 chars Auto Scaling guidance Info Select this if you intend to use this template with EC2 Auto Scaling Provide guidance to help me set up a template that I can use with EC2 Auto Scaling Template tags No template tags are currently applied to this template. Add a template tag to apply it to the launch template. Add new tag You can add up to 50 more tags. Source template You can optionally specify a source template if you would like to create a template from another existing template Launch template name Select a launch template Launch template contents Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template. Application and OS Images (Amazon Machine Image) Info An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose Browse more AMIs. Recents My AMIs Quick Start Don't include in launch template Owned by me Shared with me Amazon

Machine Image (AMI) task-management-ami ami-0b0af13bfd54d1bc8 2025-12-25T10:39:12.000Z Virtualization: hvm ENA enabled: true Root device type: ebs Boot mode: uefi-preferred Description - Architecture x86_64 AMI ID ami-0b0af13bfd54d1bc8 Instance type Info | Get advice Advanced Instance type t3.micro Free tier eligible Family: t3 2 vCPU 1 GiB Memory Current generation: true On-Demand Ubuntu Pro base pricing: 0.0139 USD per Hour On-Demand SUSE base pricing: 0.0104 USD per Hour On-Demand Linux base pricing: 0.0104 USD per Hour On-Demand RHEL base pricing: 0.0392 USD per Hour On-Demand Windows base pricing: 0.0196 USD per Hour All generations Compare instance types Additional costs apply for AMIs with pre-installed software Key pair (login) Info You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance. Key pair name taskManagement Create new key pair Network settings Info Subnet Info Don't include in launch template Create new subnet When you specify a subnet, a network interface is automatically added to your template. Availability Zone Info Don't include in launch template Enable additional zones Firewall (security groups) Info A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance. Select existing security group Create security group Security groups Info Select security groups Compare security group rules Advanced network configuration Storage (volumes) Info EBS Volumes Hide details Volume 1 (AMI Root) : 8 GiB, EBS, General purpose SSD (gp3), 3000 IOPS AMI Volumes are not included in the template unless modified Add new volume The selected AMI contains instance store volumes, however the instance does not allow any instance store volumes. None of the instance store volumes from the AMI will be accessible from the instance Resource tags Info No resource tags are currently included in this template. Add a resource tag to include it in the launch template. Add new tag You can add up to 50 more tags. Advanced details Info Summary Software Image (AMI) task-management-ami ami-0b0af13bfd54d1bc8 Virtual server type (instance type) t3.micro Firewall (security group) - Storage (volumes) 1 volume(s) - 8 GiB Cancel Create launch template You can add up to 50 more tags. now guide me which to select option to create template

Create launch template Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions. Launch template name and description Launch template name - required task-management-template Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'. Template version description Max 255 chars Auto Scaling guidance Info Select this if you intend to use this template with EC2 Auto Scaling Provide guidance to help me set up a template that I can use with EC2 Auto Scaling Template tags No template tags are currently applied to this template. Add a template tag to apply it to the launch template. Add new tag You can add up to 50 more tags. Source template You can optionally specify a source template if you would like to create a template from another existing template Launch template name Select a launch template Launch template contents Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template. Application and OS Images (Amazon Machine Image) Info An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose Browse more AMIs. Recents My AMIs Quick Start Don't include in launch template Owned by me Shared with me Amazon Machine Image (AMI) task-management-ami ami-0b0af13bfd54d1bc8 2025-12-25T10:39:12.000Z Virtualization: hvm ENA enabled: true Root device type: ebs Boot mode: uefi-preferred Description - Architecture x86_64 AMI ID ami-0b0af13bfd54d1bc8 Instance type Info | Get advice Advanced Instance type t3.micro Free tier eligible Family: t3 2 vCPU 1 GiB Memory Current generation: true On-Demand Ubuntu Pro base pricing: 0.0139 USD per Hour On-Demand SUSE base pricing: 0.0104 USD per Hour On-Demand Linux base pricing: 0.0104 USD per Hour On-Demand RHEL base pricing: 0.0392 USD per Hour On-Demand Windows base pricing: 0.0196 USD per Hour All generations Compare instance types Additional costs apply for AMIs with pre-installed software Key pair (login) Info You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance. Key pair name taskManagement Create new key pair Network settings Info Subnet Info Don't include in launch template Create new subnet When you specify a subnet, a network interface is automatically added to your template. Availability Zone Info Don't include in launch template Enable additional zones Firewall (security groups) Info A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance. Select existing security group Create security group Security groups Info Select security groups Compare security group rules Advanced network configuration Storage (volumes) Info EBS Volumes Hide details Volume 1 (AMI Root) : 8 GiB, EBS, General purpose SSD (gp3), 3000 IOPS AMI Volumes are not included in the template unless modified Add new volume The selected AMI contains instance store volumes, however the instance does not allow any instance store volumes. None of the instance store volumes from the AMI will be accessible from the instance Resource tags Info No resource tags are currently included in this template. Add a resource tag to include it in the launch template. Add new tag You can add up to 50 more tags. Advanced details Info Summary Software Image (AMI) task-management-ami ami-0b0af13bfd54d1bc8 Virtual server type (instance type) t3.micro Firewall (security group) - Storage (volumes) 1 volume(s) - 8 GiB Cancel Create launch template You can add up to 50 more tags.

terminate ? i will not terminate the instance i have worked alot in that instance if it not worked my whole work will be wasted tell me some other way to test it

i have changes the as desired capacity to 2 and max to 3 now i have total 4 instance in my ec2 instance one is my original and other i don't is this normal behaviour ?