

Getting Started: Local Development

This guide covers the setup process for running **DevBlog** in a Windows environment using `nvm-windows` and Docker/Local MongoDB.

Prerequisites

- **Hardware Virtualization:** Enabled in your BIOS/UEFI settings (required for Docker Desktop).
 - **NVM for Windows:** [Download here](#) to manage Node versions.
 - **MongoDB:** Either [MongoDB Community Server](#) installed locally or a containerized instance.
-

Step-by-Step Local Setup

1. Environment & Node.js

First, ensure you are using the correct Node runtime as specified in the project requirements:

PowerShell

```
# Install Node v19.9.0
nvm install 19.9.0

# Switch to the version
nvm use 19.9.0

# Verify installation
node -v # Should output v19.9.0
```

2. Install Dependencies

Clone the repository and install the Node modules:

Bash

```
git clone https://github.com/tiusoro/devblog.git
cd devblog
npm install
```

3. Database Configuration

The application relies on a MongoDB connection string provided via environment variables.

1. Create a `.env` file in the root directory.
2. Add your connection string (replace with your local URI if different):

Ini, TOML

```
MONGO_URI=mongodb://127.0.0.1:27017/devblog
```

Note: If you are using Node 17+, prefer `127.0.0.1` over `localhost` in your URI to avoid IPv6 resolution issues (`ECONNREFUSED ::1`).

4. Verification & Running

The project includes a manual verification step in `utils/db.js` to ensure the handshake is successful.

To start the development server:

Bash

```
npm start
```

Upon a successful start, you should see:

```
Connected to MongoDB
```

in your terminal.



Docker Workflow (Optional)

If you prefer running via Docker Desktop:

1. Ensure **Virtualization** is enabled in BIOS.
2. Run `docker-compose up --build`.
3. If the app container starts before the DB is ready, we use `netcat-openbsd` to poll the port status.



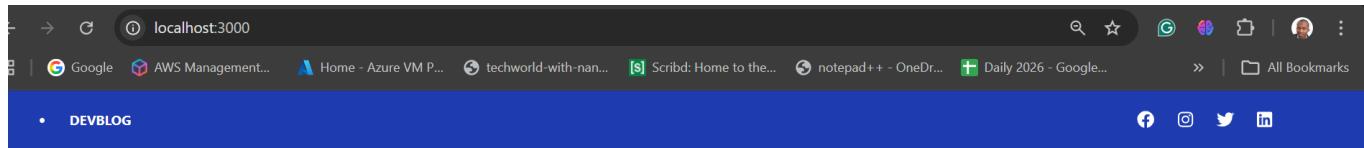
Admin Access

To create your first administrative user for the dashboard, run:

Bash

```
node create-user-cli.js
```

Screenshots



DEVBLOG

Lorem Ipsum Dolor Sit Amet

No Posts Found

DevBlog

f t i n



Local Node and Local MongoDB.

A screenshot of the Compass MongoDB interface. On the left, the 'CONNECTIONS' sidebar shows a connection to 'localhost:27017' with several databases listed: admin, config, devblog (which is expanded to show posts and users), ecommerce-shop, full-mern-movie-stack-backend, local, and test. The main panel shows the 'users' collection under the 'devblog' database. It has 0 documents, 1 index, and 0 validation rules. A query builder at the top right shows a single document with the following fields:

```
_id: ObjectId('697baf3b2a0e0486bf35d37')
name : "anthony"
email : "tiusoro@gmail.com"
password : "0245e1cbce408d6c9bdf0d58808659d6a74349fb49793e8313ad402ffc4acbef"
__v : 0
```

User created on Local MongoDB

DEVBLOG

Lorem Ipsum Dolor Sit Amet

[Add Post](#)[Logout](#)

#	Title	Subtitle	author	Date
---	-------	----------	--------	------

[← Previous](#)[Next →](#)

DevBlog

User successfully logged in

DEVBLOG

Lorem Ipsum Dolor Sit Amet

ENOENT: no such file or directory, open 'C:\Users\HomePC\Documents\Cloud-Adventures-2026\Sherifdeen\devblog\public\images\1769714149996.png'

DevBlog

Created public/images folder

[Best smile pictures](#)By [anthony](#). Published on 29/01/26[Best smile.. Continue Reading ➔](#)[1](#)

DevBlog

Created first blog post

The screenshot shows the MongoDB Compass interface. The left sidebar lists databases: 'shop' and 'wue-stack-backend'. The main area shows the 'devblog' database with a 'posts' collection. A single document is selected, showing the following data:

```

_id: ObjectId('697bb47abc1f404c8cdb30c4')
title: "Best smile pictures"
subtitle: "Best smile"
content: "This is a sample of a good passport for your linkedin profile"
author: "anthony"
slug: "best-smile-pictures"
image: "/images/1769714810590.webp"
createdAt: 2026-01-29T19:26:50.607+00:00
__v: 0

```

Checked database for post

Decided to move it one step docker by moving the DB to docker while the nodejs is still local and achieving this with docker build and docker run.

1. Created Dockerfiles for mongo and mongo-express
2. inserted in folders in the structure

mongo-docker-app/

```

    |
    └── mongo/
        └── Dockerfile

    |
    └── mongo-express/
        └── Dockerfile

```

3. Did docker build and docker run for mongodb

```

# Dockerfile for mongodb
FROM mongo:7

ENV MONGO_INITDB_ROOT_USERNAME=admin
ENV MONGO_INITDB_ROOT_PASSWORD=admin123
ENV MONGO_INITDB_DATABASE=appdb

EXPOSE 27017

```

```
ENV MONGO_INITDB_ROOT_PASSWORD=anndh  
C:\Users\HomePC\Documents\Cloud-Adventures-2026\Sherifdeen\devblog\models  
U 7 EXPOSE 27017  
U 8  
U 9  
  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS + ⋮ | [ ]  
U Tony's-PC:devblog (main)$ docker build -t devblog-mongo ./mongo-docker-app/mongo  
U => [1/1] FROM docker.io/library/mongo:7@sha256:606f8e029603330411a7dd10b5ffd50eefc297fc80cee89f10a 3.6s  
U => => sha256:606f8e029603330411a7dd10b5ffd50eefc297fc80cee89f10a455e496a76ae7 3.25kB / 3.25kB 0.0s  
U => => sha256:153d075ff9e0cc36f0be1c48df4d74a01b1f94c5dd249eb994f7ec99862ce88 2.48kB / 2.48kB 0.0s  
U => => sha256:1eca0e3663c3af7947a57444cc8fd76963f08a51518b50f3ec4c466abb579e2b 7.88kB / 7.88kB 0.0s  
M => => sha256:bb7081c6342108e9b2653cbdbf993a6900b50688a646d775e2295154c4cfcd2 1.51MB / 1.51MB 1.8s  
U => => sha256:595688014fa6af0be74b640a9294314944348b03d61d48734de27b56083bbe6 1.78kB / 1.78kB 1.1s  
U => => sha256:6f4ebca3e823b18dac366f72e537b1772bc3522a5c7ae299d6491fb17378410e 3.15MB / 29.54MB 3.3s  
U => => sha256:65ba831633afe7da109f3baa57276238c62ab9e56e071b925c7e5185fd0ae7a9 898.10kB / 898.10kB 3.0s  
U => => sha256:d4c85d6ccca97c15a4fb55aca9b549cc750f2fc4168c722b7bb24b6b61a9ea45 116B / 116B 2.6s  
U => => sha256:76027ee736179df0bbf47af52181b2b23d7158c75fb8e70ccf1d47a37641783d 0B / 262B 3.3s  
U => => sha256:1e537dd4932788fba85bc789544478b5e16a31c9902f340ef9f08aa2d0dfa8a8 0B / 251.84MB 3.3s
```

4. Did same for mongo-express

```
# Docker build for mongodb  
docker build -t devblog-mongo ./mongo-docker-app/mongo  
  
# Docker run for mongodb  
docker run -d \  
  --name mongo-db \  
  -p 27017:27017 \  
  devblog-mongo
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS + ⋮  
U Tony's-PC:devblog (main)$ docker build -t devblog-mongo ./mongo-docker-app/mongo  
U [+]: Building 120.0s (5/5) FINISHED docker:desktop-linux  
U - SecretsUsedInArgOrEnv: Do not use ARG or ENV instructions for sensitive data (ENV "MONGO_INITDB_ROOT_PASSWORD") (line 4)  
  
M View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/w490yo5kc644c5pj8z40mn39  
U ● Tony's-PC:devblog (main)$ docker run -d \  
U   --name mongo-db \  
U   -p 27017:27017 \  
U   devblog-mongo  
U   bad6fd46859bf053e24aa6894e5482143fb27aa9330855a0b12cb1335fb93f9  
U ○ Tony's-PC:devblog (main)$
```

Did Docker run for mongo-express

```
# Docker build for mongo-express  
docker build -t devblog-mongo-express ./mongo-docker-app/mongo-express  
  
# Docker run for mongo-express
```

```
docker run -d \
--name devblog-mongo-express \
-p 8081:8081 \
--link mongo-db \
devblog-mongo-express
```

5. Voila - Mongo-express

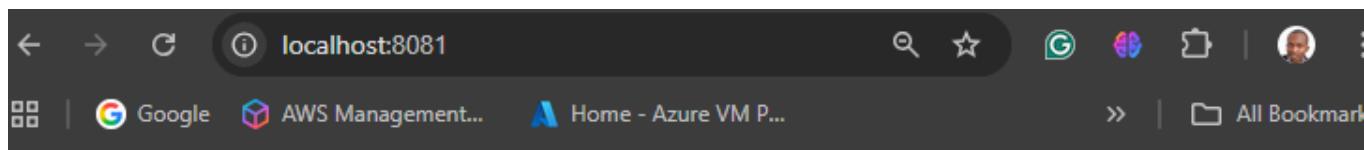
Mongo Express Database

Databases		Database Name	Create Database
<button>View</button>	admin	<button>Del</button>	
<button>View</button>	config	<button>Del</button>	
<button>View</button>	local	<button>Del</button>	

Server Status			
Hostname	bad6fd46859b	MongoDB Version	7.0.29
Uptime	820 seconds	Node Version	18.20.3

6. Edited .env file

```
MONGO_URI=mongodb://admin:admin123@localhost:27017/devblog?authSource=admin
```



Mongo Express

Databases	Database Name	Create Database
<button>View</button>	admin	<button>Del</button>
<button>View</button>	config	<button>Del</button>
<button>View</button>	devblog	<button>Del</button>
<button>View</button>	local	<button>Del</button>

Server Status



7. Move to create user
8. did node create-user-cli.js

Note: Noted there was connection success blocking the Enter name. Commenting it out at db.js

```
U PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

M ⚡ Tonys-PC:devblog (main)$ node create-user-cli.js
U [dotenv@17.2.3] injecting env (1) from .env -- tip: ⚡ load multiple .env files with { path: ['.env.local
U ', '.env'] }
M Enter name: Connection Success
M ⚡ Tonys-PC:devblog (main)$ node create-user-cli.js
U [dotenv@17.2.3] injecting env (1) from .env -- tip: ⚡ add observability to secrets: https://dotenvx.com/o
U ps
U Enter name: Anthony
U Enter email: tiusoro@gmail.com
U Enter password: tony2011
U User created successfully!
diamond Tonys-PC:devblog (main)$
```

 Delete all 1 documents retrieved

_id	name	email	password
  697c1c2474253c91f707b9dd	Anthony	tiusoro@gmail.com	0245e1cbce408d6c9bdf0d58808659d6a74349fb4

9. Connected to localhost:3000/admin. Logged in with the user and added a new post



[Brand new laptops for sale](#)

By [Anthony](#), Published on 30/01/26

Going at good discount prices
DevBlog

10. Checked to see it at the mongodb using mongo-express

Mongo Express Database: devblog Collection: posts

Simple Advanced

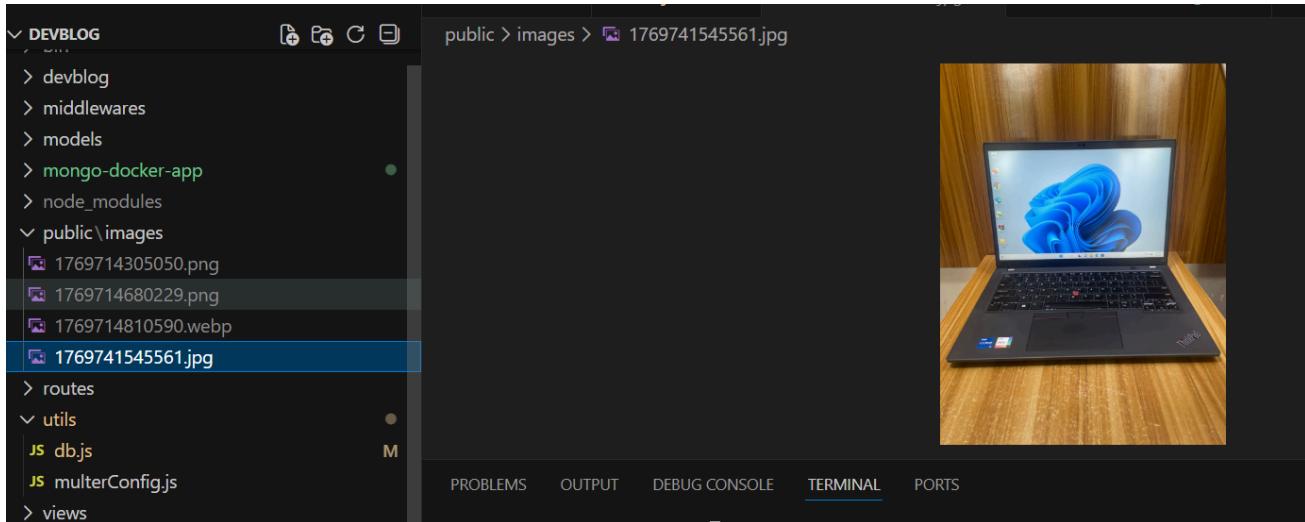
Key Value

String Find

Delete all 1 documents retrieved

_id	title	subtitle	content	author	slug	image
🔗 697c1ce95bb65dcb6e08447f	Brand new laptops for sale	New laptops	Going at good discount prices	Anthony	brand-new-laptops-for-sale	/images/1769741545561.jpg

11. Checked to see the image in public/images/folder



Full mode Dockerfile and Docker compose

1. Created dockerfile for the local nodejs

Inorder to create public/images folder in the docker image before building we need to create a bash script called start.sh.

start.sh

```
```bash
#!/usr/bin/env bash
create ./public/images directory if it doesnt exist
if [! -d "public/images/"]; then
 mkdir -p ./public/
 echo "Created directory: ./public/images"
else
 echo "Directory already exists: ./public/images"
fi

npm run start
```

#### Dockerfile for the nodejs application

```
FROM node:19-alpine
WORKDIR /app
COPY package.json ./
RUN npm install
COPY ..
```

```
EXPOSE 3000
ENTRYPOINT ["./start.sh"]
```

## Docker compose.yml file

```
version: '3.8'
name: devblog-app
services:

 web: # => service name
 build: .
 ports:
 - "8000:3000"
 env_file:
 - .env
 depends_on:
 - mongo
 restart: always

 mongo:
 image: mongo:latest
 restart: always
 environment:
 MONGO_INITDB_ROOT_USERNAME: admin
 MONGO_INITDB_ROOT_PASSWORD: admin123

 mongo-express:
 image: mongo-express
 restart: always
 ports:
 - 8081:8081
 environment:
 ME_CONFIG_MONGODB_SERVER: mongo
 ME_CONFIG_BASICAUTH_ENABLED: true
 ME_CONFIG_BASICAUTH_USERNAME: user
 ME_CONFIG_BASICAUTH_PASSWORD: pass

 depends_on:
 - mongo
```

## Issues:

On running docker compose up...it brings up the mongo express and mongodb but keeps

hinting on problem with the nodejs specifically at the entrypoint start.sh file. The file has been given executable permission and it is pasted above but wouldnt bring up the devblog container.

```
mongo-1 | {"t":{"$date":"2026-01-30T13:37:39.549+00:00"},"s":":","c":"NETWORK","id":22944,"ctx":"conn35","msg":"Connection ended","attr":{"remote":"172.19.0.3:43504","isLoadBalanced":false,"uuid":{"uuid":{"$uuid":"9effffccb-1e8a-4361-8c71-ba03b2442e89"}}, "connectionId":35,"connectionCount":1}}
mongo-1 | {"t":{"$date":"2026-01-30T13:37:39.549+00:00"},"s":":","c":"NETWORK","id":22944,"ctx":"conn34","msg":"Connection ended","attr":{"remote":"172.19.0.3:43488","isLoadBalanced":false,"uuid":{"uuid":{"$uuid":"b3dfc546-77f7-4193-8600-a1c922a74f51"}}, "connectionId":34,"connectionCount":0}}
mongo-express-1 exited with code 1 (restarting)
web-1 exited with code 255 (restarting)
web-1 | exec ./start.sh: no such file or directory
```

**Edited the docker-compose.yml file and nodejs application to use CMD instead of ENTRYPOINT.**

### Dockerfile

```
FROM node:19-alpine
WORKDIR /app
COPY package.json .
RUN npm install
COPY . .
EXPOSE 3000
CMD ["npm", "start"]
```

### .env

```
MONGO_URI=mongodb://admin:admin123@mongo:27017/devblog?authSource=admin
```

### docker-compose.yml

```
version: '3.8'
name: devblog-app
services:
 web: # => service name
 build: .
 ports:
 - "8000:3000"
```

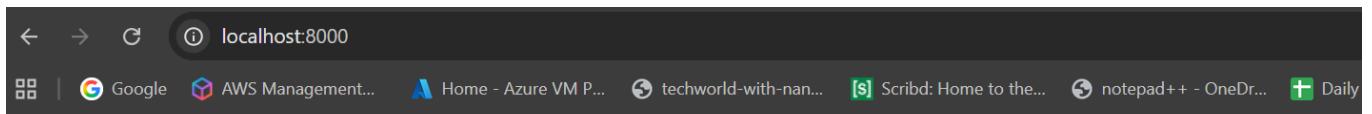
```
env_file:
 - .env
depends_on:
 - mongo
restart: always

mongo:
 image: mongo:4.0
 restart: always
 environment:
 MONGO_INITDB_ROOT_USERNAME: admin
 MONGO_INITDB_ROOT_PASSWORD: admin123

mongo-express:
 image: mongo-express
 restart: always
 ports:
 - 8081:8081
 environment:
 ME_CONFIG_MONGODB_SERVER: mongo
 ME_CONFIG_MONGODB_ADMINUSERNAME: admin
 ME_CONFIG_MONGODB_ADMINPASSWORD: admin123
 ME_CONFIG_BASICAUTH_ENABLED: true
 ME_CONFIG_BASICAUTH_USERNAME: user
 ME_CONFIG_BASICAUTH_PASSWORD: pass

 depends_on:
 - mongo
```

Voila! it worked



Lorem Ipsum Dolor Sit Amet

## No Posts Found

DevBlog

<http://localhost:8000>



## Mongo Express

Databases		Database Name	+ Create Database
	admin		Del
	config		Del
	devblog		Del
	local		Del

<https://localhost:8081> mongodb showing devblog

Added post and pic to see if it will appear on the database

<http://localhost:8000/admin>

localhost:8000/admin/#

Google AWS Management... Home - Azure VM P... techworld-with-nan... [S]

- [DevBlog](#)



[DevBlog](#)

Lorem Ipsum Dolor Sit Amet

[Add Post](#) [Logout](#)

# Title Subtitle author Date

1 Best vacation spot best vacation Anthony 30/01/26 [Edit](#) [Delete](#)



[Best vacation spot](#)

By [Anthony](#), Published on 30/01/26

[best vacation...](#) [Continue Reading ➔](#)

1  
DevBlog

Voila! its shows at mongodb database

Key	Value	Type	Find																		
<span style="color: red;">⚠ Delete all 1 documents retrieved</span>																					
<table border="1"><thead><tr><th>_id</th><th>title</th><th>subtitle</th><th>content</th><th>author</th><th>slug</th><th>image</th><th>createdAt</th><th>_v</th></tr></thead><tbody><tr><td><a href="#"></a> <a href="#"></a> 697d095eaa9b304a362c33c3</td><td>Best vacation spot</td><td>best vacation</td><td>vacation spots</td><td>Anthony</td><td>best-vacation-spot</td><td>/images/1769802078314.webp</td><td>Fri Jan 30 2026 19:41:18 GMT+0000 (Coordinated Universal Time)</td><td>0</td></tr></tbody></table>				_id	title	subtitle	content	author	slug	image	createdAt	_v	<a href="#"></a> <a href="#"></a> 697d095eaa9b304a362c33c3	Best vacation spot	best vacation	vacation spots	Anthony	best-vacation-spot	/images/1769802078314.webp	Fri Jan 30 2026 19:41:18 GMT+0000 (Coordinated Universal Time)	0
_id	title	subtitle	content	author	slug	image	createdAt	_v													
<a href="#"></a> <a href="#"></a> 697d095eaa9b304a362c33c3	Best vacation spot	best vacation	vacation spots	Anthony	best-vacation-spot	/images/1769802078314.webp	Fri Jan 30 2026 19:41:18 GMT+0000 (Coordinated Universal Time)	0													

Rename Collection

Image appears on the files of the docker container

Name	Note	Size	Last modified	Mode
1769714305050.png		737 Bytes	1 day ago	-rwxr-xr-x
1769714680229.png		1.3 MB	1 day ago	-rwxr-xr-x
1769714810590.webp		38.3 kB	1 day ago	-rwxr-xr-x
1769741545561.jpg		108.2 kB	17 hours ago	-rwxr-xr-x
1769802078314.webp	ADDED	378.2 kB	6 minutes ago	-rw-r--r--
README.md		1.2 kB	6 days ago	-rwxr-xr-x
routes			7 days ago	drwxr-xr-x
start.sh		273 Bytes	7 hours ago	-rwxr-xr-x

Did docker compose down and lost all the post and images

## Persisting data and images by creating docker volumes

```
version: '3.8'

name: devblog-app

services:
 web:
 build: .
 ports:
 - "8000:3000"
 env_file:
 - .env
 depends_on:
 - mongo
 restart: always
 volumes:
 - web-images:/app/public/images # Moved here so the app can use it

 mongo:
 image: mongo:4.0
 restart: always
 environment:
 MONGO_INITDB_ROOT_USERNAME: admin
 MONGO_INITDB_ROOT_PASSWORD: admin123
 volumes:
```

```

- mongo-data:/data/db

mongo-express:
 image: mongo-express
 restart: always
 ports:
 - "8081:8081"
 environment:
 ME_CONFIG_MONGODB_SERVER: mongo
 ME_CONFIG_MONGODB_ADMINUSERNAME: admin
 ME_CONFIG_MONGODB_ADMINPASSWORD: admin123
 ME_CONFIG_BASICAUTH_ENABLED: "true"
 ME_CONFIG_BASICAUTH_USERNAME: user
 ME_CONFIG_BASICAUTH_PASSWORD: pass
 depends_on:
 - mongo

volumes:
 mongo-data:
 driver: local
 web-images:
 driver: local

```



[A must visit destination](#)

By [Anthony](#), Published on 30/01/26

[best vacations.. Continue Reading →](#)

[1](#)  
DevBlog

Created a new post after doing docker compose up

Did docker compose down then docker compose up and the post was still there

```
● 40 - mongo
U 41
 42 volumes:
 43 mongo-data:
 44 driver: local
 45 web-images:
 46 driver: local
 47
M 48
```



[A must visit destination](#)

By [Anthony](#), Published on 30/01/26

[best vacations.. Continue Reading →](#)

1

DevBlog

## Part 2

dc-helper.sh script

```
#!/bin/bash

DevBlog Docker Compose Helper
=====

COMMAND=$1
TARGET=$2
FOLLOW=$3

show_help() {
```

```

echo "DevBlog Docker Compose Helper"
echo "=====
echo "Usage: ./dc-helper.sh <command> [service]"
echo ""
echo "Commands:"
echo " up [service] Start all services, or a specific service"
echo " down [service] Stop all services, or a specific service"
echo " build [service] Build all services, or a specific service"
echo " restart [service] Restart all services, or a specific service"
echo " logs [-f] [service] View logs. Use -f to follow"
echo " status [service] Show status of all or a specific container"
echo " seed Create admin user (run seeding script)"
echo " shell [service] Open a shell inside a container (web,
mongo)"
echo " clean Stop containers and remove volumes
(DESTSTRUCTIVE)"
echo " help Show this help message"
echo ""
echo "Services: web, mongo, mongo-express"
}

```

```

case $COMMAND in
up)
 docker compose up -d $TARGET
 ;;
down)
 docker compose stop $TARGET
 ;;
build)
 docker compose build $TARGET
 ;;
restart)
 docker compose restart $TARGET
 ;;
logs)
 # Handle the -f flag logic
 if ["$TARGET" == "-f"]; then
 docker compose logs -f $FOLLOW
 else
 docker compose logs $TARGET
 fi
 ;;
status)
 docker compose ps $TARGET
 ;;
seed)

```

```

 echo "Seeding database..."
 docker compose exec web node create-user-cli.js
 ;;
shell)
 if ["$TARGET" == "mongo"]; then
 docker compose exec mongo mongosh -u admin -p admin123 --authenticationDatabase admin
 elif [-n "$TARGET"]; then
 docker compose exec $TARGET sh
 else
 echo "Error: Please specify a service (web or mongo)"
 fi
 ;;
clean)
 echo "⚠️ WARNING: This will remove all containers and VOLUMES (data loss)!"
 read -p "Are you sure? (y/N): " confirm
 if [[$confirm == [yY] || $confirm == [yY][eE][sS]]]; then
 docker compose down -v
 echo "Cleanup complete."
 else
 echo "Operation cancelled."
 fi
 ;;
help|*)
 show_help
 ;;
esac

```

## Part 3

chmod +x dc-helper.sh

**Test help ./dc-helper.sh help**

```

● Tonys-PC:devblog (main)$./dc-helper.sh help
DevBlog Docker Compose Helper
=====
Usage: ./dc-helper.sh <command> [service]

Commands:
 up [service] Start all services, or a specific service
 down [service] Stop all services, or a specific service
 build [service] Build all services, or a specific service
 restart [service] Restart all services, or a specific service
 logs [-f] [service] View logs. Use -f to follow
 status [service] Show status of all or a specific container
 seed Create admin user (run seeding script)
 shell [service] Open a shell inside a container (web, mongo)
 clean Stop containers and remove volumes (DESTRUCTIVE)
 help Show this help message

Services: web, mongo, mongo-express
❖ Tonys-PC:devblog (main)$

```

## Start all services ./dc-helper.sh up

```

● Tonys-PC:devblog (main)$./dc-helper.sh up
time="2026-01-30T21:31:49+01:00" level=warning msg="C:\\\\Users\\\\HomePC\\\\Documents\\\\Cloud-Adventures-2026\\\\S
herifdeen\\\\devblog\\\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please re
move it to avoid potential confusion"
[+] Running 3/3
 ✓ Container devblog-app-mongo-1 Running 0.0s
 ✓ Container devblog-app-mongo-express-1 Running 0.0s
 ✓ Container devblog-app-web-1 Running 0.0s
❖ Tonys-PC:devblog (main)$

```

## Check status ./dc-helper.sh status

```

● Tonys-PC:devblog (main)$./dc-helper.sh status
time="2026-01-30T21:33:36+01:00" level=warning msg="C:\\\\Users\\\\HomePC\\\\Documents\\\\Cloud-Adventures-2026\\\\S
herifdeen\\\\devblog\\\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please re
move it to avoid potential confusion"
NAME IMAGE COMMAND SERVICE CREATED
STATUS PORTS COMMAND SERVICE CREATED
devblog-app-mongo-1 mongo:4.0 "docker-entrypoint.s..." mongo 21 minutes ago
Up 21 minutes 27017/tcp
devblog-app-mongo-express-1 mongo-express "/sbin/tini -- /dock..." mongo-express 21 minutes ago
Up 21 minutes 0.0.0.0:8081->8081/tcp, [::]:8081->8081/tcp
devblog-app-web-1 devblog-app-web "docker-entrypoint.s..." web 21 minutes ago
Up 21 minutes 0.0.0.0:8000->3000/tcp, [::]:8000->3000/tcp
❖ Tonys-PC:devblog (main)$

```

## View logs ./dc-helper.sh logs

```
● Tonys-PC:devblog (main)$./dc-helper.sh logs
time="2026-01-30T21:35:55+01:00" level=warning msg="C:\\Users\\HomePC\\Documents\\Cloud-Adventures-2026\\S
herifdeen\\devblog\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please re
move it to avoid potential confusion"
mongo-express-1 | Waiting for mongo:27017...
mongo-express-1 | /docker-entrypoint.sh: connect: Connection refused
mongo-express-1 | /docker-entrypoint.sh: line 15: /dev/tcp/mongo/27017: Connection refused
mongo-express-1 | Fri Jan 30 20:11:50 UTC 2026 retrying to connect to mongo:27017 (2/10)
mongo-express-1 | /docker-entrypoint.sh: connect: Connection refused
mongo-express-1 | /docker-entrypoint.sh: line 15: /dev/tcp/mongo/27017: Connection refused
mongo-express-1 | Fri Jan 30 20:11:51 UTC 2026 retrying to connect to mongo:27017 (3/10)
mongo-express-1 | /docker-entrypoint.sh: connect: Connection refused
mongo-express-1 | /docker-entrypoint.sh: line 15: /dev/tcp/mongo/27017: Connection refused
mongo-express-1 | Fri Jan 30 20:11:52 UTC 2026 retrying to connect to mongo:27017 (4/10)
```

## Seed the database ./dc-helper.sh seed

```
○ Tonys-PC:devblog (main)$./dc-helper.sh seed
Seeding database...
time="2026-01-30T21:39:09+01:00" level=warning msg="C:\\Users\\HomePC\\Documents\\Cloud-Adventures-2026\\S
herifdeen\\devblog\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please re
move it to avoid potential confusion"
[dotenv@17.2.3] injecting env (0) from .env -- tip: ⚙ suppress all logs with { quiet: true }
```

## Open shells ./dc-helper.sh shell web

```
○ Tonys-PC:devblog (main)$./dc-helper.sh shell web
time="2026-01-30T21:40:42+01:00" level=warning msg="C:\\Users\\HomePC\\Documents\\Cloud-Adventures-2026\\S
herifdeen\\devblog\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please re
move it to avoid potential confusion"
/app #
/app #
/app # █
```

## Restart a service ./dc-helper.sh restart web

```
● Tonys-PC:devblog (main)$./dc-helper.sh restart web
time="2026-01-30T21:42:12+01:00" level=warning msg="C:\\Users\\HomePC\\Documents\\Cloud-Adventures-2026\\S
herifdeen\\devblog\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please re
move it to avoid potential confusion"
[+] Restarting 1/1
 ✓ Container devblog-app-web-1 Started
2.1s
❖ Tonys-PC:devblog (main)$ █
```

## Clean up ./dc-helper.sh clean

```
● Tonys-PC:devblog (main)$./dc-helper.sh clean
⚠ WARNING: This will remove all containers and VOLUMES (data loss)!
Are you sure? (y/N): y
time="2026-01-30T21:43:25+01:00" level=warning msg="C:\\Users\\HomePC\\Documents\\Cloud-Adventures-2026\\S
herifdeen\\devblog\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please re
move it to avoid potential confusion"
[+] Running 6/6
 ✓ Container devblog-app-mongo-express-1 Removed 1.0s
 ✓ Container devblog-app-web-1 Removed 1.7s
 ✓ Container devblog-app-mongo-1 Removed 1.2s
 ✓ Network devblog-app_default Removed 0.9s
 ✓ Volume devblog-app_web_images Removed 0.2s
 ✓ Volume devblog-app_mongo_data Removed 0.1s
Cleanup complete.
❖ Tonys-PC:devblog (main)$ █
```

# Deployed to AWS Ubuntu VM

A screenshot of a web browser window. The address bar shows "Not secure 34.228.165.12:8000". The page title is "DEVBLOG". The main content area displays the "DEVBLOG" logo and the placeholder text "Lorem Ipsum Dolor Sit Amet". Below this, a red banner with the text "No Posts Found" is visible. The browser interface includes various tabs and icons at the top.

## DEVBLOG

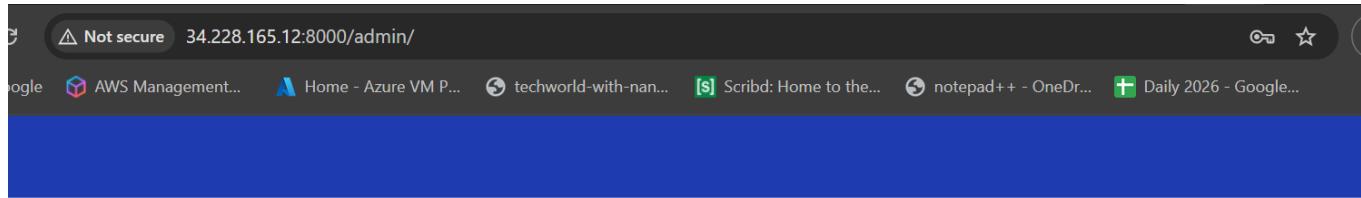
Lorem Ipsum Dolor Sit Amet

No Posts Found

A screenshot of a web browser window. The address bar shows "Not secure 34.228.165.12:8081". The page title is "Mongo Express Database". The main content area displays the "Mongo Express" logo and the heading "Mongo Express". Below this, a table titled "Databases" lists four databases: "admin", "config", "devblog", and "local". Each database row has a "View" button in a green box and a "Del" button in a red box. A search bar labeled "Database Name" and a "Create Database" button are also present. The browser interface includes various tabs and icons at the top.

Databases	
<button>View</button>	admin
<button>View</button>	config
<button>View</button>	devblog
<button>View</button>	local

logged in as a user



# DEVBLOG

Lorem Ipsum Dolor Sit Amet

Add Post

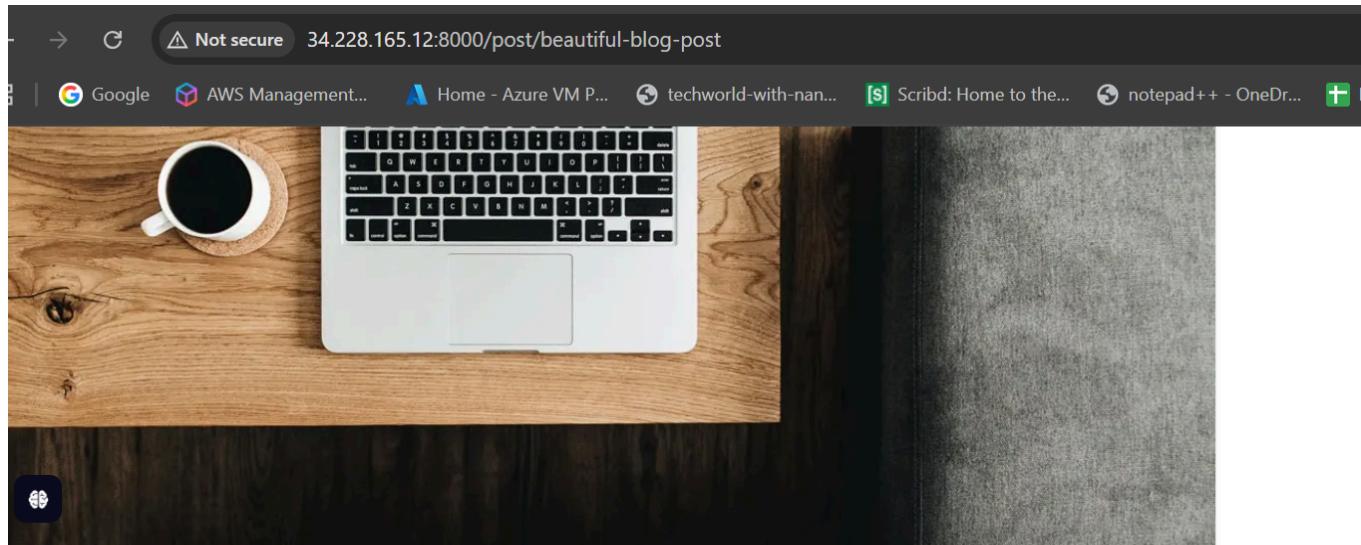
Logout

#	Title	Subtitle	author	Date
---	-------	----------	--------	------

← Previous

Next →

Added a post and check it out on the landing page and database



## Beautiful Blog Post

By Anthony Usoro, Published on 31/01/26

## These Are The 10 Most Beautiful Blog Designs In The World

It's no secret that blogs are the pillar of any successful company website. A frequently updated blog will drive traffic to the brand, guide users with targeted content, and act as a vital tool to engage with

→ ⌂ Not secure 34.228.165.12:8081/db/devblog/posts

Google AWS Management... Home - Azure VM P... techworld-with-nan... Scribd: Home to the... notepad++ - OneDr... Daily 2026 - Google...

Mongo Express Database: devblog Collection: posts

## Viewing Collection: posts

[New Document](#) [New Index](#)

[Simple](#) [Advanced](#)

Key Value String [Find](#)

[Delete all 1 documents retrieved](#)

_id	title	subtitle	content	author	slug	image	createdAt
697d6c277f981d50b7eaf75f	Beautiful Blog Post	Best Post so far	<h1 class="entry-title ds-typography__h1 ds-color...	Anthony Usoro	beautiful-blog-post	/images/1769827366487.webp	Sat Jan 31 2026 02:42:47 GMT+0000 (Coordinated Universal Time)

→ ⌂ Not secure 34.228.165.12:8081/db/devblog/users

Google AWS Management... Home - Azure VM P... techworld-with-nan... Scribd: Home to the... notepad++ - OneDr... Daily 2026 - Google...

Mongo Express Database: devblog Collection: users

## Viewing Collection: users

[New Document](#) [New Index](#)

[Simple](#) [Advanced](#)

Key Value String [Find](#)

[Delete all 1 documents retrieved](#)

_id	name	email	password
697d6b4e632073454629b12b	Anthony Usoro	tiusoro@gmail.com	0245e1cbce408d6c9bdf0d58808659d6a74349fb49793e831...