

Deploy a Professional Website (Confidence Project)

Objective:

The goal of this project is to deploy a simple, professional marketing/portfolio website using Nginx on an Ubuntu VM, following real-world DevOps practices such as pre-flight checks, validation and step-by-step deployment.

This project focuses on confidence in fundamentals, not complexity.

Environment Readiness Checklist:

Before starting the deployment, the following prerequisites were verified to ensure the system is production-ready.

- Ubuntu Virtual Machine:
 - Using an Ubuntu VM(same instance used in previous assignment:
Deploy a React App on Ubuntu VM Using Nginx)
 - Nginx Installed and Running
 - Internet Access Verified
 - VM Inbound Rules allow HTTP (Port 80)
- Since, all the readiness checks were successfully verified in earlier assignments, the same Ubuntu VM instance is reused for this deployment to maintain environment consistency.

Note: If you need to checkout the previous assignment, checkout the link below:

 [Deploy a React App on Ubuntu VM Using Nginx](#)

Task 0 - Pre-flight Check (Mandatory):

- The following mandatory pre-flight checks are:

A. Host Identification:

```
$ hostname
```

```
ubuntu@ip-172-31-7-210:~$ hostname  
ip-172-31-7-210 ←  
ubuntu@ip-172-31-7-210:~$  
ubuntu@ip-172-31-7-210:~$
```

Venkatesh Gangavarapu

- Hostname Identity confirmed.

B. OS Verification:

```
$ lsb_release -a  
ubuntu@ip-172-31-7-210:~$ lsb_release -a  
No LSB modules are available.  
Distributor ID: Ubuntu  
Description:    Ubuntu 24.04.3 LTS  
Release:        24.04  
Codename:       noble  
ubuntu@ip-172-31-7-210:~$
```

Venkatesh Gangavarapu

- OS Verified.

C. Nginx Installation Verification:

```
$ nginx -v
```

```
ubuntu@ip-172-31-7-210:~$ nginx -v  
nginx version: nginx/1.24.0 (Ubuntu)
```

Venkatesh Gangavarapu

- Nginx installation verified.

D. Nginx health Check:

```
$ sudo systemctl status nginx --no-pager
```

```
ubuntu@ip-172-31-7-210:~$ sudo systemctl status nginx --no-pager  
● nginx.service - A high performance web server and a reverse proxy server  
  Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)  
  Active: active (running) since Wed 2026-01-21 14:11:16 UTC; 19h ago  
    Docs: man:nginx(8)  
  Process: 30038 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)  
  Process: 30040 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)  
 Main PID: 30041 (nginx)  
   Tasks: 3 (limit: 1008)  
  Memory: 2.6M (peak: 3.1M)  
    CPU: 100ms  
   CGroup: /system.slice/nginx.service  
         ├─30041 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"  
         ├─30042 "nginx: worker process"  
         ├─30043 "nginx: worker process"  
  
Jan 21 14:11:16 ip-172-31-7-210 systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...  
Jan 21 14:11:16 ip-172-31-7-210 systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.  
ubuntu@ip-172-31-7-210:~$
```

Venkatesh Gangavarapu

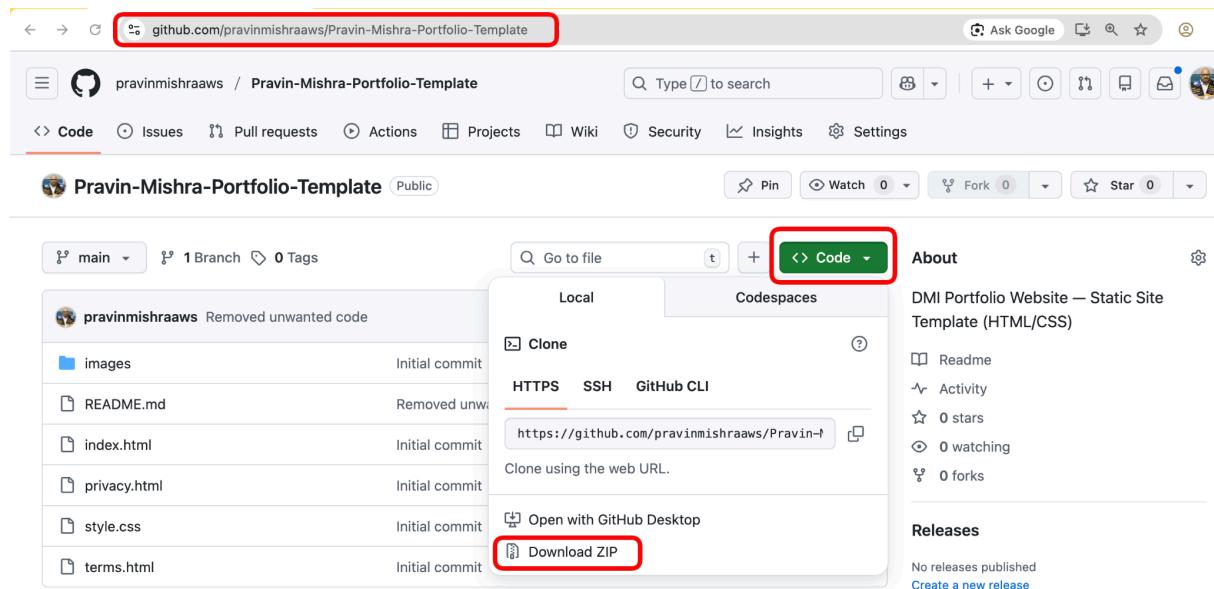
- Nginx health confirmed active(running).

Result: All checks passed successfully, confirming the server is ready for website deployment.

Task 1 - Get the Website Source Code

- The first step in deploying the professional website is to obtain the source code from GitHub.

Step 1: Refer to the below reference image provided for guidance on the repository layout and files.



Step 2: Navigate to the GitHub repository using the link below:

GitHub Repository:

<https://github.com/pravinmishraaws/Pravin-Mishra-Portfolio-Template/tree/main>

Step 3: Review the repository structure to understand the website files before deployment.

Step 4: Download or clone the repository to the Ubuntu VM in preparation for serving it via Nginx.

1. Clone the GitHub Repository and verify

```
$ git clone
https://github.com/pravinmishraaws/Pravin-Mishra-Portfolio-Template.git
```

```
ubuntu@ip-172-31-7-210:~$ git clone https://github.com/pravinmishraaws/Pravin-Mishra-Portfolio-Template.git
Cloning into 'Pravin-Mishra-Portfolio-Template'...
remote: Enumerating objects: 25, done. →
remote: Counting objects: 100% (25/25), done.
remote: Compressing objects: 100% (20/20), done.
remote: Total 25 (delta 6), reused 23 (delta 4), pack-reused 0 (from 0)
Receiving objects: 100% (25/25), 3.86 MiB | 3.68 MiB/s, done.
Resolving deltas: 100% (6/6), done.
ubuntu@ip-172-31-7-210:~$ ls -l
total 4
drwxrwxr-x 4 ubuntu ubuntu 4096 Jan 22 10:25 Pravin-Mishra-Portfolio-Template
```

Venkatesh Gangavarapu

(Or)

2. Download the zip file link in to VM

```
$ wget
https://github.com/pravinmishraaws/Pravin-Mishra-Portfolio-Template/archive/refs/heads/main.zip
```

```

ubuntu@ip-172-31-7-210:~$ ls -l
total 0
ubuntu@ip-172-31-7-210:~$ wget https://github.com/pravinmishraaws/Pravin-Mishra-Portfolio-Template/archive/refs/heads/main.zip
--2026-01-22 10:30:06-- https://github.com/pravinmishraaws/Pravin-Mishra-Portfolio-Template/archive/refs/heads/main.zip
Resolving github.com (github.com)... 20.207.73.82
Connecting to github.com (github.com)|20.207.73.82|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://code.load.github.com/pravinmishraaws/Pravin-Mishra-Portfolio-Template/zip/refs/heads/main [following]
--2026-01-22 10:30:06-- https://code.load.github.com/pravinmishraaws/Pravin-Mishra-Portfolio-Template/zip/refs/heads/main
Resolving code.load.github.com (code.load.github.com)... 20.207.73.88
Connecting to code.load.github.com (code.load.github.com)|20.207.73.88|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/zip]
Saving to: 'main.zip'

main.zip
[ <=>
2026-01-22 10:30:07 (20.0 MB/s) - 'main.zip' saved [4049122]

ubuntu@ip-172-31-7-210:~$ ls -l
total 3956
-rw-rw-r-- 1 ubuntu ubuntu 4049122 Jan 22 10:30 main.zip
ubuntu@ip-172-31-7-210:~$ ubuntu@ip-172-31-7-210:~$
```

Venkatesh Gangavarapu

3. Extract the zip file

`$ unzip # If not unzip not installed follow below`

`$ sudo apt install unzip`

```

ubuntu@ip-172-31-7-210:~$ unzip main.zip
Command 'unzip' not found, but can be installed with:
sudo apt install unzip
ubuntu@ip-172-31-7-210:~$ ubuntu@ip-172-31-7-210:~$ sudo apt install unzip
Reading package lists... done
Building dependency tree... done
Reading state information... done
Suggested packages:
  zip
The following NEW packages will be installed:
  unzip
0 upgraded, 1 newly installed, 0 to remove and 46 not upgraded.
Need to get 174 kB of archives.
After this operation, 384 kB of additional disk space will be used.
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 unzip amd64 6.0-28ubuntu4.1 [174 kB]
Fetched 174 kB in 0s (11.1 MB/s)
Selecting previously unselected package unzip.
(Reading database ... 142000 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-28ubuntu4.1_amd64.deb ...
Unpacking unzip (6.0-28ubuntu4.1) ...
Setting up unzip (6.0-28ubuntu4.1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-7-210:~$ |
```

Venkatesh Gangavarapu

`$ unzip main.zip`

```

ubuntu@ip-172-31-7-210:~$ 
ubuntu@ip-172-31-7-210:~$ unzip main.zip
Archive: main.zip
c84824056eed17ee00c2bcd8cc04a16ea9950adc
  creating: Pravin-Mishra-Portfolio-Template-main/
  inflating: Pravin-Mishra-Portfolio-Template-main/README.md
  creating: Pravin-Mishra-Portfolio-Template-main/images/
  inflating: Pravin-Mishra-Portfolio-Template-main/images/Devops.jpg
  inflating: Pravin-Mishra-Portfolio-Template-main/images/Git.jpg
  inflating: Pravin-Mishra-Portfolio-Template-main/images/awsCloud.jpg
  inflating: Pravin-Mishra-Portfolio-Template-main/images/dmi-course.jpg
  inflating: Pravin-Mishra-Portfolio-Template-main/images/image.png
  extracting: Pravin-Mishra-Portfolio-Template-main/images/logo.png
  inflating: Pravin-Mishra-Portfolio-Template-main/images/profile.jpg
  inflating: Pravin-Mishra-Portfolio-Template-main/images/signature.png
  inflating: Pravin-Mishra-Portfolio-Template-main/index.html
  inflating: Pravin-Mishra-Portfolio-Template-main/privacy.html
  inflating: Pravin-Mishra-Portfolio-Template-main/style.css
  inflating: Pravin-Mishra-Portfolio-Template-main/terms.html
ubuntu@ip-172-31-7-210:~$ 
ubuntu@ip-172-31-7-210:~$ 
ubuntu@ip-172-31-7-210:~$ ls -l
total 3960
drwxrwxr-x 3 ubuntu ubuntu    4096 Jan 16 04:15 Pravin-Mishra-Portfolio-Template-main
-rw-rw-r-- 1 ubuntu ubuntu 4049122 Jan 22 12:17 main.zip
ubuntu@ip-172-31-7-210:~$ 
ubuntu@ip-172-31-7-210:~$ 
ubuntu@ip-172-31-7-210:~$ 
ubuntu@ip-172-31-7-210:~$ 
ubuntu@ip-172-31-7-210:~$ 

```

Venkatesh Gangavarapu

- We can view the unzip folder of main.zip

Task 2 — Add Ownership Proof (Anti-Copy Change)

To ensure originality and prevent blind copying, an ownership proof must be added to the website footer. This change clearly identifies who deployed the website, when and under which cohort.

- Navigate to the folder and modify the index.html file to change the Footer Line.

```

$ cd Pravin-Mishra-Portfolio-Template-main/
ubuntu@ip-172-31-7-210:~$ cd Pravin-Mishra-Portfolio-Template-main/
ubuntu@ip-172-31-7-210:~/Pravin-Mishra-Portfolio-Template-main$ 
ubuntu@ip-172-31-7-210:~/Pravin-Mishra-Portfolio-Template-main$ ls -l
total 68
-rw-rw-r-- 1 ubuntu ubuntu 1045 Jan 16 04:15 README.md
drwxrwxr-x 2 ubuntu ubuntu 4096 Jan 16 04:15 images
-rw-rw-r-- 1 ubuntu ubuntu 20687 Jan 16 04:15 index.html
-rw-rw-r-- 1 ubuntu ubuntu 4427 Jan 16 04:15 privacy.html
-rw-rw-r-- 1 ubuntu ubuntu 17218 Jan 16 04:15 style.css
-rw-rw-r-- 1 ubuntu ubuntu 4542 Jan 16 04:15 terms.html
ubuntu@ip-172-31-7-210:~/Pravin-Mishra-Portfolio-Template-main$ 
ubuntu@ip-172-31-7-210:~/Pravin-Mishra-Portfolio-Template-main$ | 

```

Venkatesh Gangavarapu

- Edit the index.html file and add below lines

```

<p>Crafted with <span>cloud</span> excellence by Pravin
Mishra</p>
<p><strong>Deployed by:</strong> DMI Cohort 2 | Venkatesh
Gangavarapu | Group 4 | Week 1 | 16-01-2026</p>

```

```
$ vi index.html
  <p>© <span id="year"></span> Pravin Mishra. All rights reserved.</p>
  <p>Crafted with <span>cloud</span> excellence by Pravin Mishra</p>
  <p><strong>Deployed by:</strong> DMI Cohort 2 | Venkatesh Gangavarapu | Group 6 | Week 2 | 22-01-2026</p>
</div>

```

Venkatesh Gangavarapu

Why This Matters:

- Confirms deployment ownership
- Prevents reuse of unchanged copies
- Mirrors real-world practices where deployment accountability matters
- Makes the project traceable and auditable

Task 3 — Deploy Website via Nginx (Production-Like Steps)

This task focuses on deploying the website using Nginx in a way that closely resembles how static websites are served in production environments.

Step 1: Verify Current Nginx Web Root:

- Before deploying new content, verify the existing web root directory.

```
$ ls -lah /var/www/html
ubuntu@ip-172-31-7-210:~$ ls -lah /var/www/html/
total 76K
drwxr-xr-x 3 www-data www-data 4.0K Jan 22 12:14 .
drwxr-xr-x 3 root      root     4.0K Jan 21 14:09 ..
-rw-r--r-- 1 root      root     1.1K Jan 22 11:13 README.md
drwxr-xr-x 2 root      root     4.0K Jan 22 11:13 images
-rw-r--r-- 1 root      root     24K Jan 22 12:14 index.html
-rw-r--r-- 1 root      root     4.4K Jan 22 11:13 privacy.html
-rw-r--r-- 1 root      root     17K Jan 22 11:13 style.css
-rw-r--r-- 1 root      root     4.5K Jan 22 11:13 terms.html
ubuntu@ip-172-31-7-210:~$
ubuntu@ip-172-31-7-210:~$
ubuntu@ip-172-31-7-210:~$ |
```

Venkatesh Gangavarapu

- This ensures you know what content is currently being served and avoids accidental overwrites.

Step 2: Backup Existing Content (Safety First)

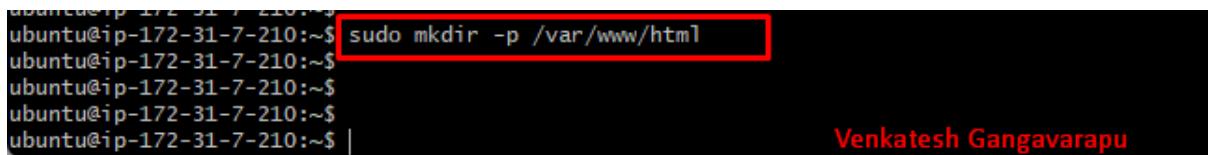
- Create a backup of the current web root before deploying the new website.

```
$ sudo mv /var/www/html /var/www/html_backup_$(date +%F)
ubuntu@ip-172-31-7-210:~$ sudo mv /var/www/html/ /var/www/html_backup_$(date +%F)
ubuntu@ip-172-31-7-210:~$ ls /var/www/
html_backup_2026-01-22
ubuntu@ip-172-31-7-210:~$ ubuntu@ip-172-31-7-210:~$ |
```

Venkatesh Gangavarapu

- Create a Nginx web root folder again

```
$ sudo mkdir -p /var/www/html
```



The terminal window shows the command `sudo mkdir -p /var/www/html` highlighted with a red box. The output of the command is visible below it.

Venkatesh Gangavarapu

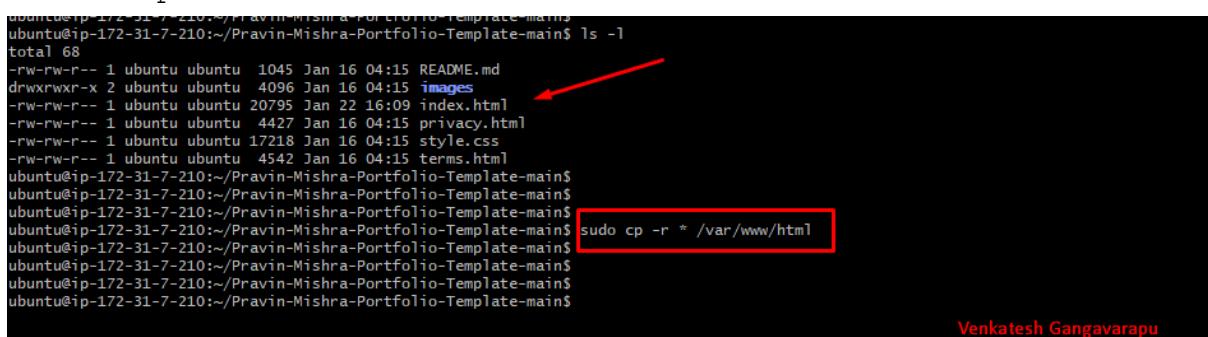
Why this matters:

- Backing up content allows fast rollback in case the deployment fails.

Step 3: Copy Website Source Code to Web Root

- Navigate to the source directory and copy the files to the Nginx web root

```
$ sudo cp -r * /var/www/html
```



The terminal window shows the command `sudo cp -r * /var/www/html` highlighted with a red box. The output of the command is visible below it. A red arrow points to the `index.html` file in the `ls -l` output.

Venkatesh Gangavarapu

- Ensure you are inside the website source directory before running this command

Why this matters:

- Nginx serves content from `/var/www/html` by default, copying files here makes them accessible to users.

Step 4: Set Correct Permissions and Ownership

- Set ownership and permissions to the Nginx web root folder

```
$ sudo chown -R www-data:www-data /var/www/html
```

```
ubuntu@ip-172-31-7-210:/var/www/html$ ls -lth
total 68K
-rw-r--r-- 1 root root 4.5K Jan 22 16:25 terms.html
-rw-r--r-- 1 root root 17K Jan 22 16:25 style.css
-rw-r--r-- 1 root root 4.4K Jan 22 16:25 privacy.html
-rw-r--r-- 1 root root 21K Jan 22 16:25 index.html
drwxr-xr-x 2 root root 4.0K Jan 22 16:25 images
-rw-r--r-- 1 root root 1.1K Jan 22 16:25 README.md
ubuntu@ip-172-31-7-210:/var/www/html$ sudo chown -R www-data:www-data /var/www/html
ubuntu@ip-172-31-7-210:/var/www/html$ ls -lth
total 68K
-rw-r--r-- 1 www-data www-data 4.5K Jan 22 16:25 terms.html      Before Owner Change
-rw-r--r-- 1 www-data www-data 17K Jan 22 16:25 style.css
-rw-r--r-- 1 www-data www-data 4.4K Jan 22 16:25 privacy.html
-rw-r--r-- 1 www-data www-data 21K Jan 22 16:25 index.html
drwxr-xr-x 2 www-data www-data 4.0K Jan 22 16:25 images
-rw-r--r-- 1 www-data www-data 1.1K Jan 22 16:25 README.md
ubuntu@ip-172-31-7-210:/var/www/html$ ls
ubuntu@ip-172-31-7-210:/var/www/html$ |                                         After Owner Change
                                         Venkatesh Gangavarapu
```

```
$ sudo chmod -R 755 /var/www/html
```

```
ubuntu@ip-172-31-7-210:/var/www/html$ 
ubuntu@ip-172-31-7-210:/var/www/html$ sudo chmod 755 /var/www/html
ubuntu@ip-172-31-7-210:/var/www/html$ ls -lth
total 68K
-rw-r--r-- 1 www-data www-data 4.5K Jan 22 16:25 terms.html
-rw-r--r-- 1 www-data www-data 17K Jan 22 16:25 style.css
-rw-r--r-- 1 www-data www-data 4.4K Jan 22 16:25 privacy.html
-rw-r--r-- 1 www-data www-data 21K Jan 22 16:25 index.html
drwxr-xr-x 2 www-data www-data 4.0K Jan 22 16:25 images
-rw-r--r-- 1 www-data www-data 1.1K Jan 22 16:25 README.md
ubuntu@ip-172-31-7-210:/var/www/html$ 
ubuntu@ip-172-31-7-210:/var/www/html$ |                                         Venkatesh Gangavarapu
```

Why this matters:

- Correct permissions prevent access issues and follow security best practices.

Step 5: Verify Nginx Configuration

- Before restarting Nginx, validate the configuration

```
$ sudo nginx -t
ubuntu@ip-172-31-7-210:/var/www/html$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
ubuntu@ip-172-31-7-210:/var/www/html$ 
ubuntu@ip-172-31-7-210:/var/www/html$ 
ubuntu@ip-172-31-7-210:/var/www/html$ |                                         Venkatesh Gangavarapu
```

Why this matters:

- This prevents downtime caused by invalid configurations.

Step 6: Restart Nginx to Apply Changes

```
$ sudo systemctl restart nginx
```

```
ubuntu@ip-172-31-7-210:/var/www/html$ sudo systemctl restart nginx
ubuntu@ip-172-31-7-210:/var/www/html$
```

Venkatesh Gangavarapu

```
$ sudo systemctl status nginx --no-pager
```

```
ubuntu@ip-172-31-7-210:/var/www/html$ sudo systemctl status nginx --no-pager
● nginx.service - A high performance web server and a reverse proxy server
  Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
  Active: active (running) since Thu 2026-01-22 16:35:40 UTC; 1min 6s ago
    Docs: man:nginx(8)
   Process: 4016 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Process: 4019 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
 Main PID: 4020 (nginx)
   Tasks: 3 (limit: 1017)
  Memory: 2.3M (peak: 2.5M)
     CPU: 17ms
    CGroup: /system.slice/nginx.service
            └─4020 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
              ├─4021 "nginx: worker process"
              ├─4022 "nginx: worker process"
              ├─4023 "nginx: worker process"

Jan 22 16:35:40 ip-172-31-7-210 systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
Jan 22 16:35:40 ip-172-31-7-210 systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
```

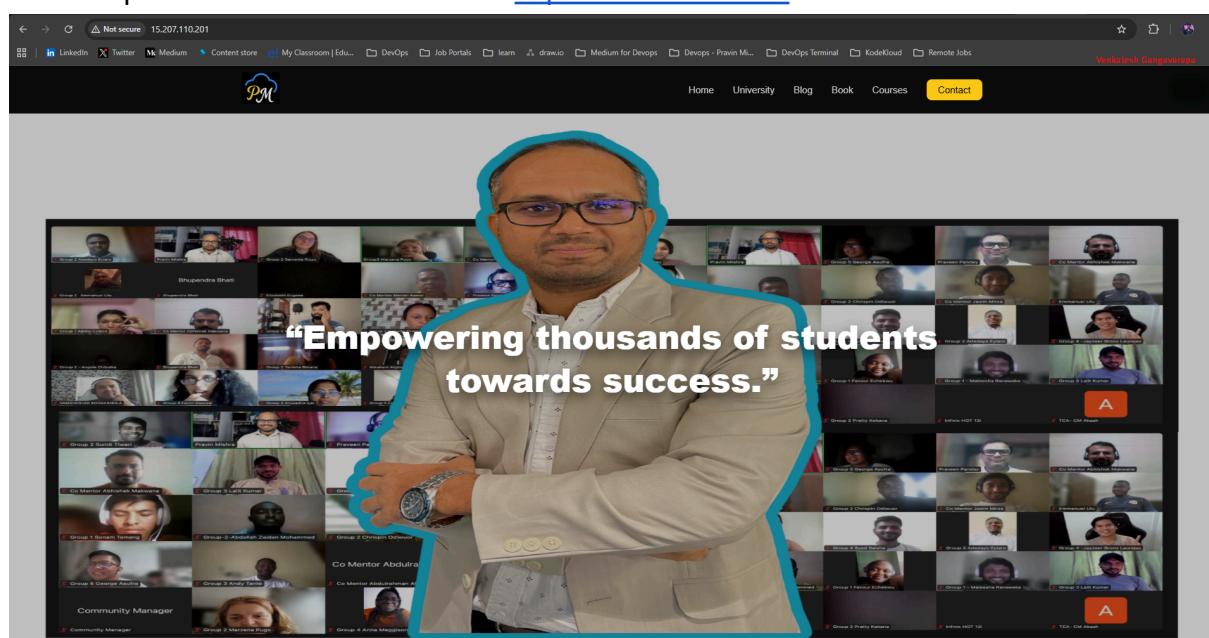
```
ubuntu@ip-172-31-7-210:/var/www/html$
```

Venkatesh Gangavarapu

- This confirm that Nginx is active (running)

Task 4 — Verify Website is Live Public

- Verify Deployment in Browser
 - Open the website in a browser: <http://15.207.110.201>



- Website loads successfully.
- Verify the footer that displays the ownership proof

Venkatesh Gangavarapu



Empowering individuals and organizations with cutting-edge Cloud and DevOps expertise through courses, books, and consulting.

Services

- Cloud Courses
- DevOps Books
- Consulting
- Corporate Training

Quick Links

- About Me
- Contact
- Privacy Policy
- Terms of Service

© Pravin Mishra. All rights reserved. Crafted with **cloud** excellence by Pravin Mishra

Deployed by: DMI Cohort 2 | Venkatesh Gangavarapu | Group 6 | Week 2 | 22-01-2026

- Footer section clearly visible with Ownership proof.

This confirms the Professional Website (Confidence Project) is deployed successfully.

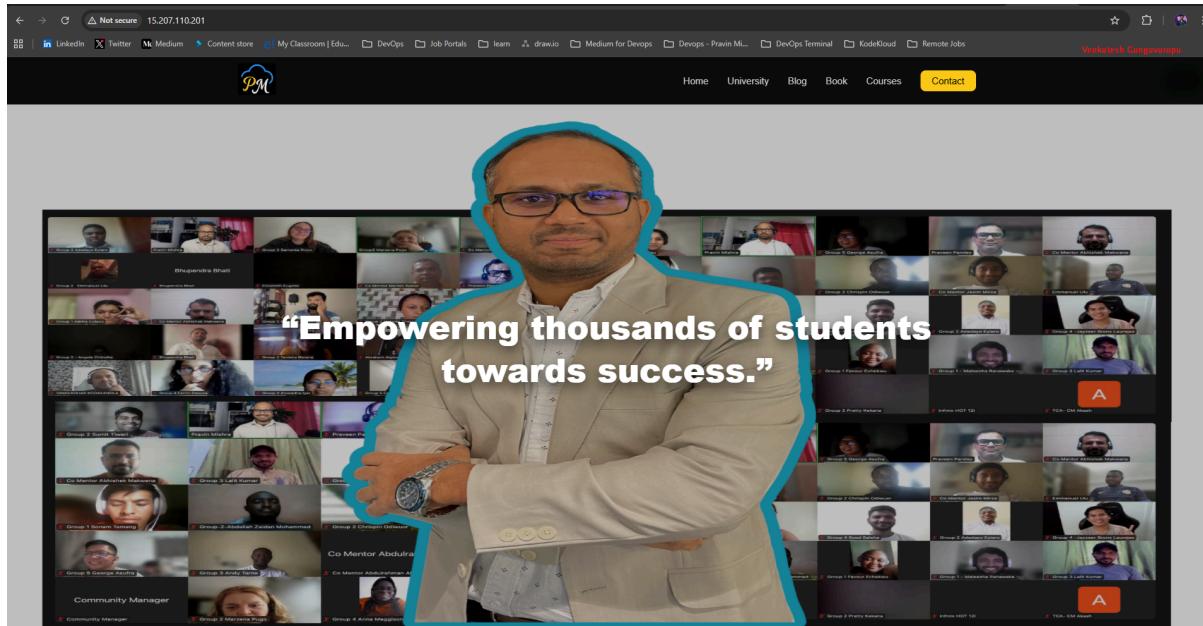
Task 5 — Mini “Real DevOps” Ops Check (Mandatory)

- This task validates that the website deployment is stable, correct and production-ready.
- The focus is not just on deployment, but on operational confidence.

OPS Validation Steps:

1. External Availability Check (User Perspective)

- <http://15.207.110.201>



- Users can access from outside the server, this means traffic is reaching the server and Nginx is serving the intended site.

2. Ownership and Release Integrity Check

- Footer display

The screenshot shows a dark-themed website footer. At the top right is the name "Venkatesh Gangavarapu". Below it are two columns: "Services" and "Quick Links". The "Services" column includes "Cloud Courses", "DevOps Books", "Consulting", and "Corporate Training". The "Quick Links" column includes "About Me", "Contact", "Privacy Policy", and "Terms of Service". In the center, there's a "Services" section with icons for LinkedIn, X (Twitter), GitHub, Email, Instagram, and YouTube. Below this is a copyright notice: "© Pravin Mishra. All rights reserved." and "Crafted with cloud excellence by Pravin Mishra". At the bottom right is a red-bordered box containing the text "Deployed by: DMI Cohort 2 | Venkatesh Gangavarapu | Group 6 | Week 2 | 22-01-2026".

- This confirms that correct content is live.

3. Configuration Safety Check (Prevents Restart Outages)

- Nginx configuration test

```
$ sudo nginx -t
ubuntu@ip-172-31-7-210:~$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok ←
nginx: configuration file /etc/nginx/nginx.conf test is successful
ubuntu@ip-172-31-7-210:~$
ubuntu@ip-172-31-7-210:~$
ubuntu@ip-172-31-7-210:~$
```

Venkatesh Gangavarapu

- This confirms the configuration is safe and no risk of outage during restart or deploy.

4. Service Health Check (systemd Discipline)

- Nginx service health check

```
$ sudo systemctl status nginx --no-pager
ubuntu@ip-172-31-7-210:~$ sudo systemctl status nginx --no-pager
● nginx.service - A high performance web server and a reverse proxy server
  Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
  Active: active (running) since Thu 2026-01-22 16:35:40 UTC; 33min ago
    Docs: man:nginx(8)
    Process: 4016 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 4019 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 4020 (nginx)
      Tasks: 3 (limit: 1017)
     Memory: 2.6M (peak: 4.0M)
        CPU: 28ms
       CGroup: /system.slice/nginx.service
           └─4020 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
              ├─4021 "nginx: worker process"
              ├─4022 "nginx: worker process"
              └─4023 "nginx: worker process"

Jan 22 16:35:40 ip-172-31-7-210 systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
Jan 22 16:35:40 ip-172-31-7-210 systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
ubuntu@ip-172-31-7-210:~$
```

Venkatesh Gangavarapu

- This confirms that Service is healthy

5. Stability Confidence Check

- Commitment

- VM will remain running for the next 24 hours
- No stop, reboot or termination

 **OPS Definition of Done:**

- Service is reachable externally
- Correct content & ownership proof is live
- Configuration is restart-safe
- Nginx service is healthy
- VM remains stable for 24 hours

Professional Website is Now Live on Ubuntu with Nginx

[Venkatesh Gangavarapu](#)

LinkedIn post URL:

https://www.linkedin.com/posts/venkatesh-gangavarapu_devops-linux-nginx-activity-7420155917125599232-s8Bz

LinkedIn Post Screenshot:

 Venkatesh Gangavarapu  Premium

Venkatesh Gangavarapu  You
DevOps Engineer | AWS • Kubernetes • Terraform • CI/CD | Linux Automation | ...
now • 

⭐ Deployed a Professional Website using Nginx on Ubuntu (Confidence Project)

As part of Week 2 assignment of DevOps Micro Internship, I successfully deployed a professional static website on an Ubuntu VM using Nginx, following production-like DevOps practices — from pre-flight checks to post-deployment OPS validation.

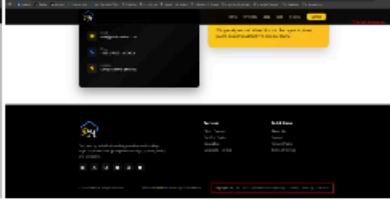
This exercise taught me that deployment is not just about making a site live, but about verifying configuration safety, service health, ownership proof, and operational confidence like a real on-call engineer.

🌐 Live Website:
👉 <http://15.207.110.201/>

Grateful for the hands-on learning and the focus on doing things the right way, not the fast way with support of [Pravin Mishra](#) and co-mentors [Manish Kumar Olajide Salami](#) and [Nkechi Anna Ahanonye](#).

★ P.S. This post is part of the DevOps Micro Internship (DMI) Cohort-2 by [Pravin Mishra](#). You can start your DevOps journey by joining this WhatsApp community: <https://lnkd.in/gCwiN5r4>.

#DevOps
#Linux
#Nginx
#ProductionReadiness
#BuildInPublic
#LearningInPublic
#WebDeployment
#DMI #PravinMishra



Like Comment Repost Send

1 impression View analytics