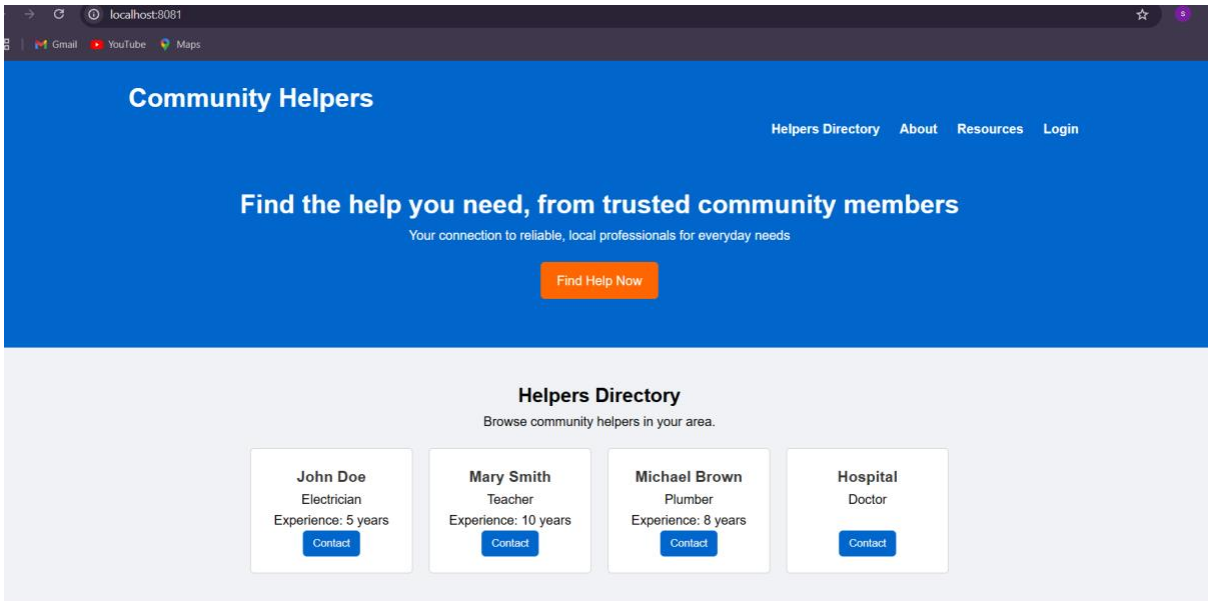
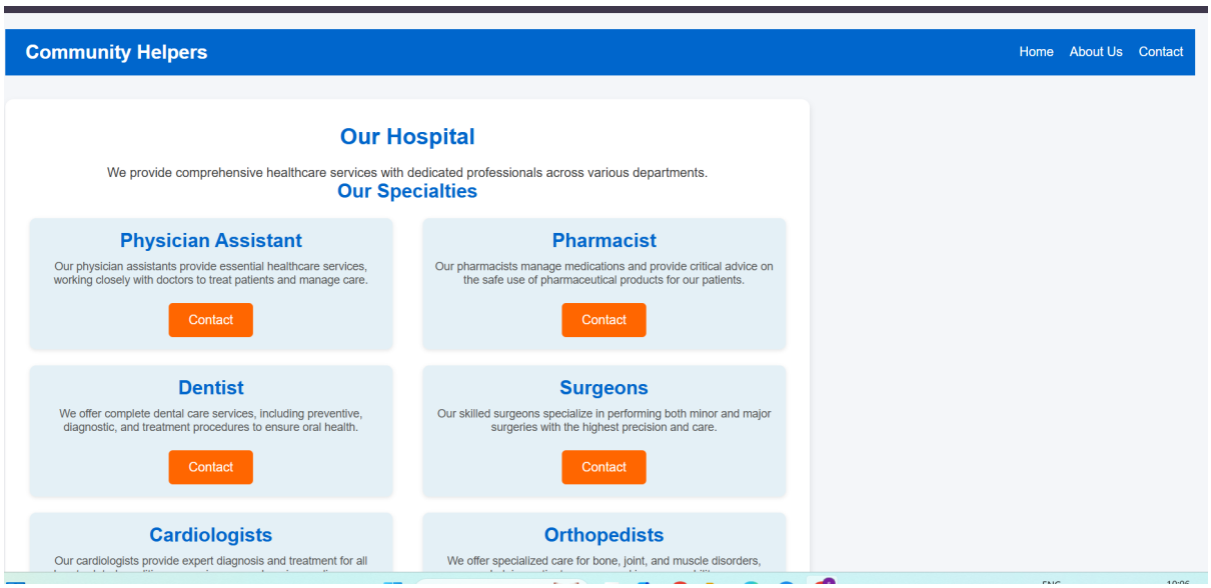
To integrate and deploy an HTML website using Docker, Jenkins, and Kubernetes, you can create a CI/CD pipeline that automates the build, test, and deployment stages. Here's a step-by-step guide to setting up a simple deployment pipeline:

Prerequisites:

1. **Docker** installed and configured on your local machine.
2. **Jenkins** server set up, with plugins for Docker, Kubernetes, and any other necessary integrations.
3. **Kubernetes cluster** (e.g., Minikube for testing, or a cloud-based cluster like EKS, GKE, or AKS).
4. **Git repository** to store your HTML website files.

WEBSITE USING HTML,CSS AND JAVASCRIPT





Step 1: Dockerize Your HTML Website

1. **Create a Dockerfile** in your HTML project's root directory to build a Docker image for the HTML website:

dockerfile

Copy code

Use a basic web server image

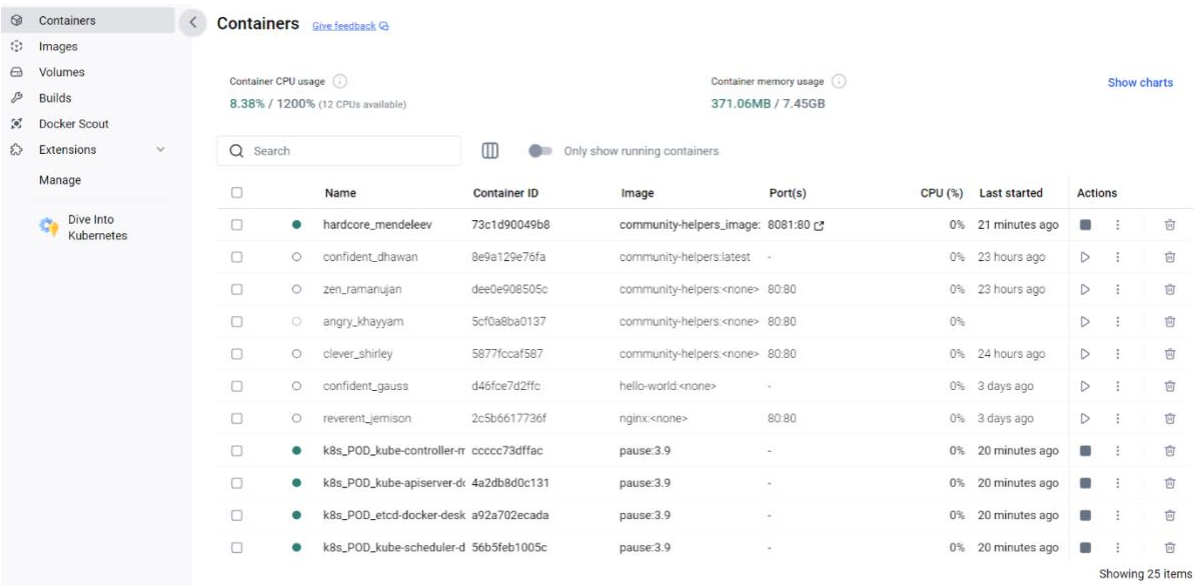
FROM nginx:alpine

Copy website files into the container

COPY . /usr/share/nginx/html

Expose port 80

EXPOSE 80



2. **Build and Test Locally**:

bash

Copy code

docker build -t my-html-website .

docker run -p 8080:80 my-html-website

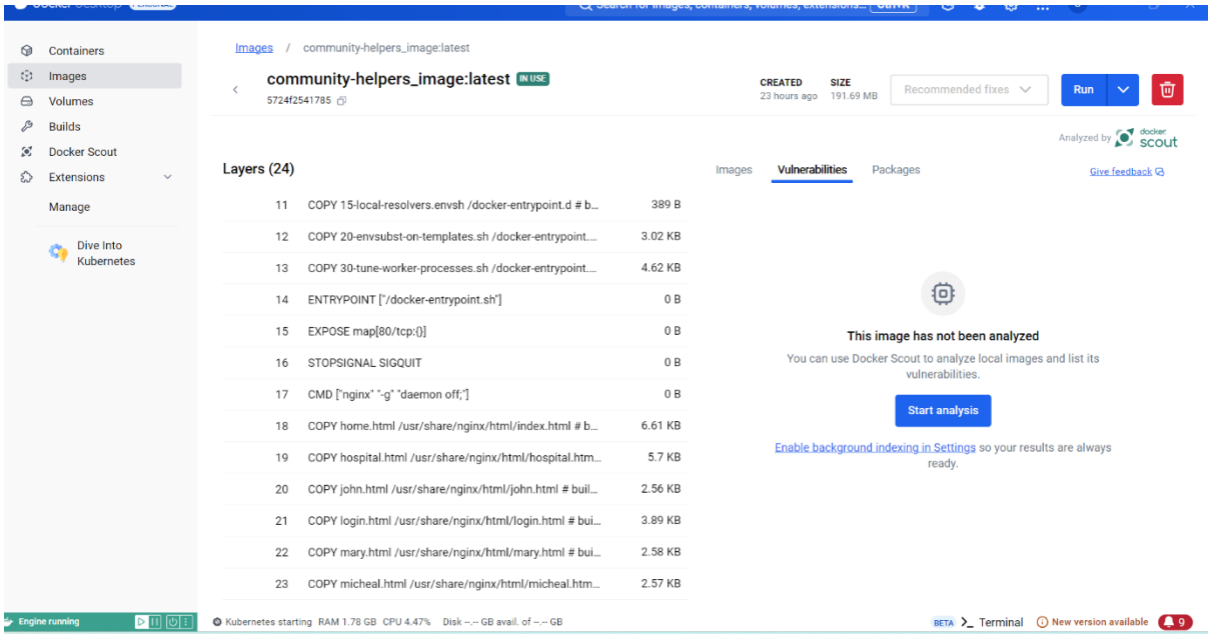
3. **Push the Image to Docker Hub** (or another registry):

bash

Copy code

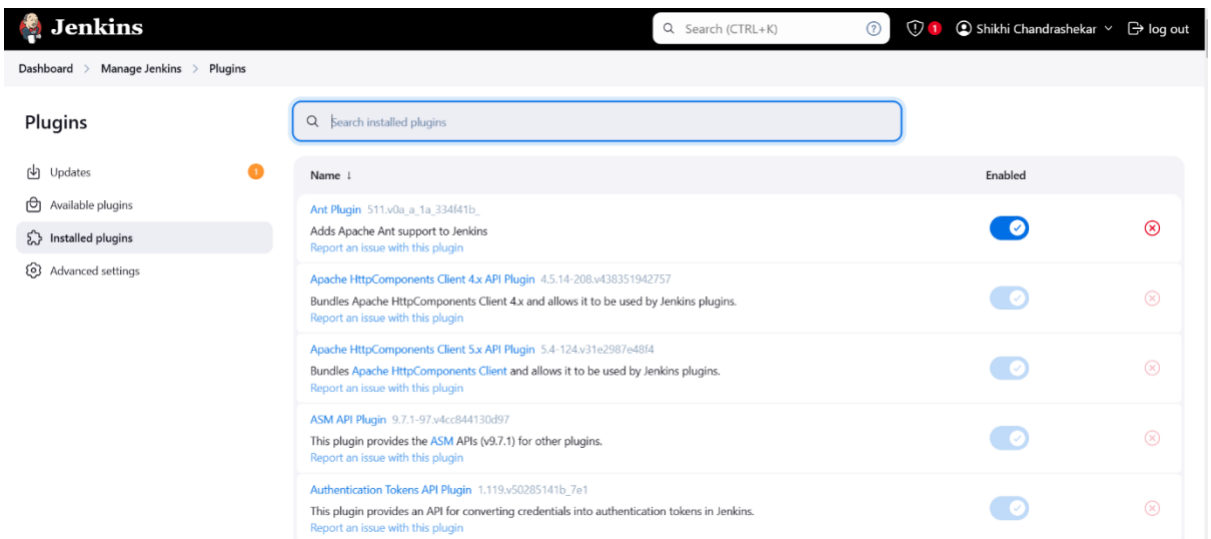
docker tag my-html-website your-dockerhub-username/my-html-website

docker push your-dockerhub-username/my-html-website

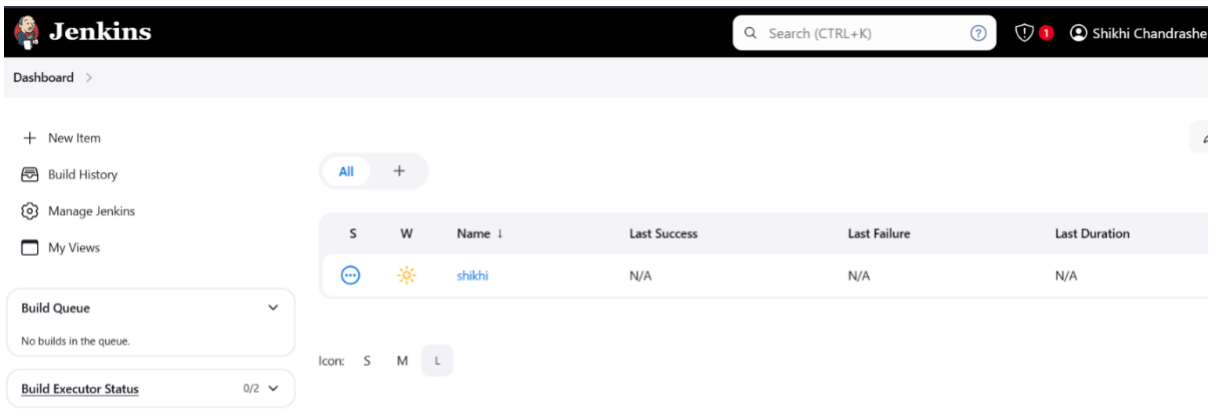


Step 2: Set Up Jenkins Pipeline

1. **Install Plugins** in Jenkins:
   - Docker Plugin
   - Kubernetes Plugin (if deploying directly to Kubernetes)
   - Pipeline Plugin



2. **Create a Jenkins Pipeline** with a Jenkinsfile: In the root of your project, create a Jenkinsfile to define the pipeline:



groovy

Copy code

```
pipeline {
  agent any
  environment {
    DOCKER_IMAGE = "your-dockerhub-username/my-html-website"
  }
  stages {
    stage('Build Docker Image') {
      steps {
        script {
          docker.build(DOCKER_IMAGE)
        }
      }
    }
    stage('Push to Docker Hub') {
      steps {
        script {
          docker.withRegistry('https://index.docker.io/v1/', 'dockerhub-credentials-id') {
```

```
                docker.image(DOCKER_IMAGE).push("latest")
            }
        }
    }
  }
  stage('Deploy to Kubernetes') {
    steps {
      kubernetesDeploy(configs: "k8s/deployment.yaml", kubeconfigId: "kubeconfig-id")
    }
  }
 }
}
```

3. **Set Up Credentials**:
    - Add Docker Hub credentials (dockerhub-credentials-id).
    - Configure Kubernetes credentials (kubeconfig-id) for deployment.

Step 3: Create Kubernetes Deployment Configuration

1. In your project directory, create a Kubernetes deployment file (k8s/deployment.yaml) to deploy the HTML website as a pod/service:

yaml

Copy code

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: html-website-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: html-website
  template:
    metadata:
```

```
    labels:
      app: html-website
  spec:
    containers:
    - name: html-website
      image: your-dockerhub-username/my-html-website:latest
      ports:
      - containerPort: 80


---
apiVersion: v1
kind: Service
metadata:
  name: html-website-service
spec:
  type: LoadBalancer
  selector:
    app: html-website
  ports:
  - protocol: TCP
    port: 80
    targetPort: 80
```

2. This file defines a **Deployment** with two replicas and a **Service** for external access.

Step 4: Run the Pipeline

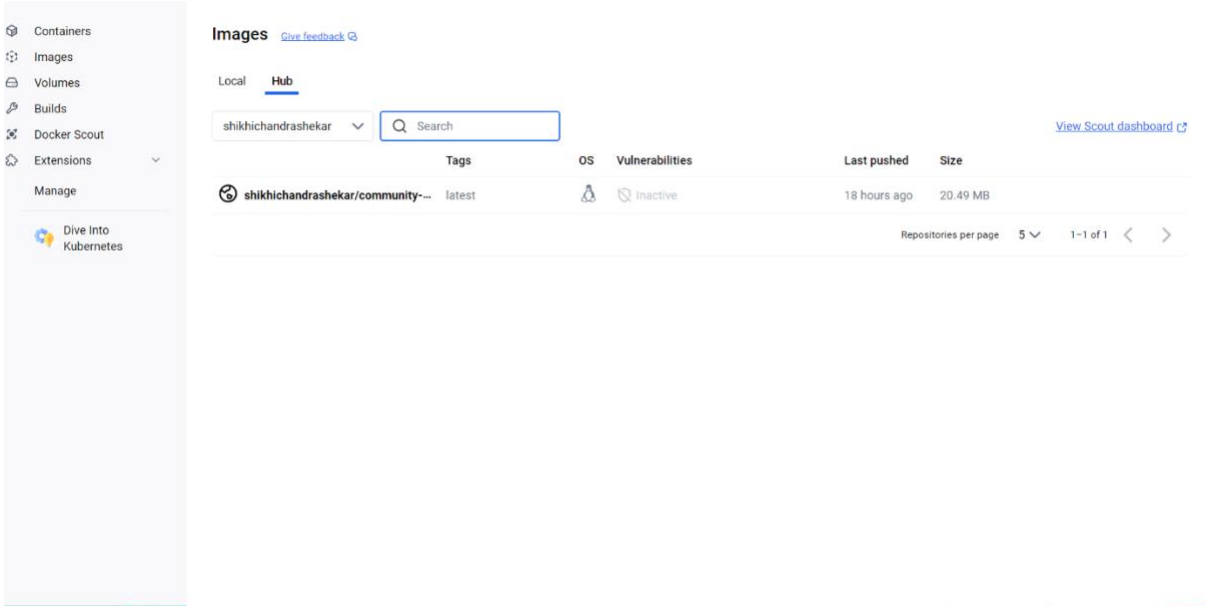
1. Commit the Jenkinsfile and k8s/deployment.yaml to your Git repository.
2. **Trigger the Jenkins Pipeline** by pushing changes or manually starting the job.
3. Jenkins will:
    - Build the Docker image.
    - Push it to Docker Hub.
    - Deploy the image to the Kubernetes cluster.

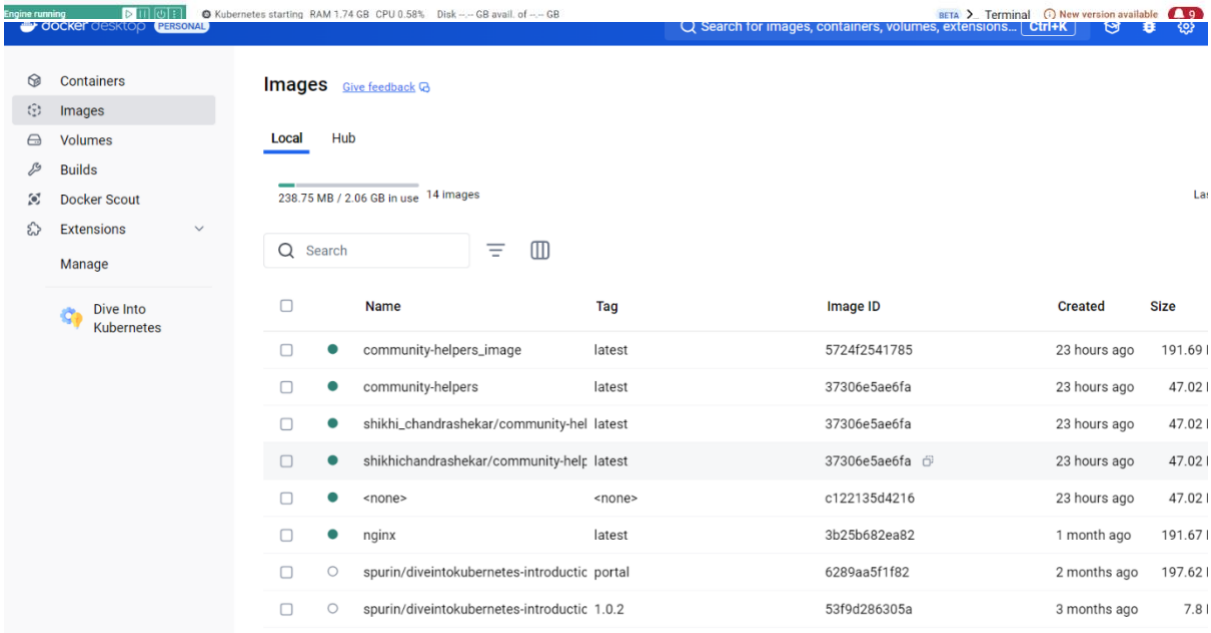
Step 5: Access the Deployed Website

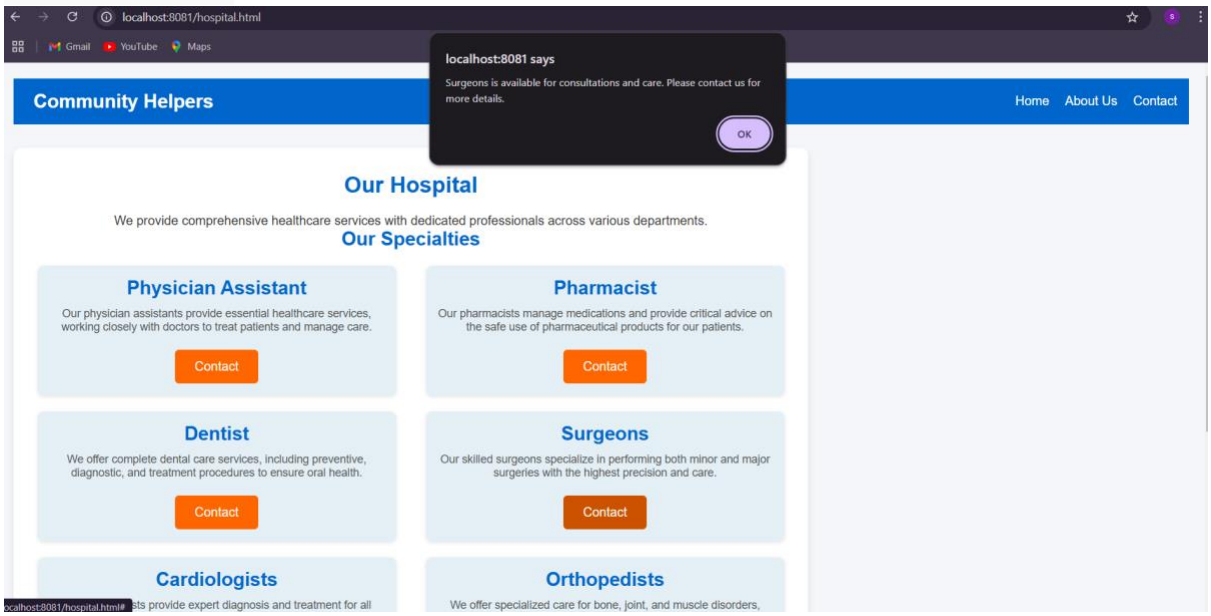
Once the deployment completes, you can access the HTML website via the IP or URL assigned to your Kubernetes Service's LoadBalancer.

This pipeline setup automates the entire process, from building and testing to deploying on Kubernetes.

4o


Containers
Images
Volumes
Builds
Docker Scout
Extensions
Manage
Dive Into Kubernetes
Images
Give feedback
Local
Hub
shikhichandrashekar
Search
View Scout dashboard
Tags
OS
Vulnerabilities
Last pushed
Size
shikhichandrashekar/community-...
latest
Inactive
18 hours ago
20.49 MB
Repositories per page
5
1-1 of 1


Engine running
Kubernetes starting RAM 1.74 GB CPU 0.58% Disk --.-- GB avail. of --.-- GB
Terminal
New version available
Search for images, containers, volumes, extensions...
Ctrl+K
Containers
Images
Volumes
Builds
Docker Scout
Extensions
Manage
Dive Into Kubernetes
Images
Give feedback
Local
Hub
238.75 MB / 2.06 GB in use
14 images
Search
Name
Tag
Image ID
Created
Size
community-helpers_image
latest
5724f2541785
23 hours ago
community-helpers
latest
37306e5ae6fa
23 hours ago
shikhi_chandrashekar/community-hel
latest
37306e5ae6fa
23 hours ago
shikhichandrashekar/community-help
latest
37306e5ae6fa
23 hours ago
<none>
<none>
c122135d4216
23 hours ago
nginx
latest
3b25b682ea82
1 month ago
spurin/diveintokubernetes-introductic
portal
6289aa5f1f82
2 months ago
spurin/diveintokubernetes-introductic
1.0.2
53f9d286305a
3 months ago


localhost:8081/hospital.html
Gmail
YouTube
Maps
localhost:8081 says
Surgeons is available for consultations and care. Please contact us for more details.
OK
Community Helpers
Home
About Us
Contact
Our Hospital
We provide comprehensive healthcare services with dedicated professionals across various departments.
Our Specialties
Physician Assistant
Our physician assistants provide essential healthcare services, working closely with doctors to treat patients and manage care.
Contact
Pharmacist
Our pharmacists manage medications and provide critical advice on the safe use of pharmaceutical products for our patients.
Contact
Dentist
We offer complete dental care services, including preventive, diagnostic, and treatment procedures to ensure oral health.
Contact
Surgeons
Our skilled surgeons specialize in performing both minor and major surgeries with the highest precision and care.
Contact
Cardiologists
Orthopedists
We offer specialized care for bone, joint, and muscle disorders,