



# Build Automated Infrastructure

Diaa Qassem  
FORTSTAK 27/07/2025

### Part 1 (30 points):

- Clone this repo: <https://github.com/Ankit6098/ToDo-List-nodejs>
- Use your own mongodb database ( check .env).
- Dockerize the application.
- Use GitHub Actions to create a CI pipeline that builds the image and pushes it to a private docker registry.

```
worker.tlp - ec2-user@54.236.204.217:22 - Bitvise xterm - ec2-user@ip-172-31-38-230:~/Fortstack/Todos-nodejs
Installed:
git-2.47.1-1.amzn2023.0.3.x86_64      git-core-2.47.1-1.amzn2023.0.3.x86_64
git-core-doc-2.47.1-1.amzn2023.0.3.noarch  perl-Error-1:0.17029-5.amzn2023.0.2.noarch
perl-File-Find-1.37-477.amzn2023.0.7.noarch  perl-Git-2.47.1-1.amzn2023.0.3.noarch
perl-TermReadKey-2.38-9.amzn2023.0.2.x86_64  perl-lib-0.65-477.amzn2023.0.7.x86_64
```

Complete!

```
[ec2-user@ip-172-31-38-230 Fortstack]$ git clone https://github.com/Ankit6098/Todos-nodejs
Cloning into 'Todos-nodejs'...
```

```
remote: Enumerating objects: 105, done.
```

```
remote: Counting objects: 100% (18/18), done.
```

```
remote: Compressing objects: 100% (6/6), done.
```

```
remote: Total 105 (delta 13), reused 12 (delta 12), pack-reused 87 (from 1)
```

```
Receiving objects: 100% (105/105), 54.00 KiB | 13.50 MiB/s, done.
```

```
Resolving deltas: 100% (47/47), done.
```

```
[ec2-user@ip-172-31-38-230 Fortstack]$ cd Todos-nodejs/
```

```
[ec2-user@ip-172-31-38-230 Todos-nodejs]$ ls -la
```

```
total 92
```

```
drwxr-xr-x. 9 ec2-user ec2-user 16384 Jul 27 18:47 .
drwxr-xr-x. 3 ec2-user ec2-user   26 Jul 27 18:47 ..
-rw-r--r--. 1 ec2-user ec2-user   89 Jul 27 18:47 .env
drwxr-xr-x. 8 ec2-user ec2-user  163 Jul 27 18:47 .git
-rw-r--r--. 1 ec2-user ec2-user   13 Jul 27 18:47 .gitignore
-rw-r--r--. 1 ec2-user ec2-user 5057 Jul 27 18:47 README.md
-rw-r--r--. 1 ec2-user ec2-user 3160 Jul 27 18:47 README.txt
drwxr-xr-x. 4 ec2-user ec2-user   27 Jul 27 18:47 assets
drwxr-xr-x. 2 ec2-user ec2-user   25 Jul 27 18:47 config
drwxr-xr-x. 2 ec2-user ec2-user  152 Jul 27 18:47 controllers
-rw-r--r--. 1 ec2-user ec2-user 2822 Jul 27 18:47 index.js
drwxr-xr-x. 2 ec2-user ec2-user   61 Jul 27 18:47 models
-rw-r--r--. 1 ec2-user ec2-user 48634 Jul 27 18:47 package-lock.json
-rw-r--r--. 1 ec2-user ec2-user   714 Jul 27 18:47 package.json
drwxr-xr-x. 2 ec2-user ec2-user   22 Jul 27 18:47 routes
drwxr-xr-x. 2 ec2-user ec2-user  107 Jul 27 18:47 views
```

```
[ec2-user@ip-172-31-38-230 Todos-nodejs]$
```

```
FROM node:18-alpine
```

```
WORKDIR /app
```

```
COPY . .
```

```
RUN npm install
```

```
CMD ["npm","start","nodemon"]
```

```
~
~
~
~
```

```
[ec2-user@ip-172-31-38-230 Todo-List-nodejs]$ cat .env
```

```
mongoDbUrl = mongodb://54.236.204.217:27017/todolistDb
```

```
[ec2-user@ip-172-31-38-230 Todo-List-nodejs]$
```

## Build Docker image

```
[ec2-user@ip-172-31-38-230 Todo-List-nodejs]$ docker build -t todo-list-nodejs .
[+] Building 9.3s (9/9) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile              0.1s
=> => transferring dockerfile: 184B                             0.0s
=> [internal] load metadata for docker.io/library/node:18-alpine 0.5s
=> [internal] load .dockerignore                                0.0s
=> => transferring context: 126B                                  0.0s
=> [1/4] FROM docker.io/library/node:18-alpine@sha256:8d6421d663b4c28fd3ebc498332f249011d118 3.7s
=> => resolve docker.io/library/node:18-alpine@sha256:8d6421d663b4c28fd3ebc498332f249011d118 0.0s
=> => sha256:ee77c6cd7c1886ecc802ad6cedef3a8ec1ea27d1fb96162bf03dd3710839b8d 6.18kB / 6.18kB 0.0s
=> => sha256:f18232174bc91741fdf3da96d85011092101a032a93a388b79e99e69c2d5c87 3.64MB / 3.64MB 0.3s
=> => sha256:8d6421d663b4c28fd3ebc498332f249011d118945588d0a35cb9bc4b8ca09d9 7.67kB / 7.67kB 0.0s
=> => sha256:929b04d7c782f04f615cf785488fed452b6569f87c73ff666ad553a7554f000 1.72kB / 1.72kB 0.0s
=> => sha256:dd71dde834b5c203d162902e6b8994cb2309ae049a0eabc4efea161b2b5a3 40.01MB / 40.01MB 1.1s
=> => sha256:1e5a4c89cee5c0826c540ab06d4b6b491c96eda01837f430bd47f0d26702d6e 1.26MB / 1.26MB 0.3s
=> => extracting sha256:f18232174bc91741fdf3da96d85011092101a032a93a388b79e99e69c2d5c870 0.5s
=> => sha256:25ff2da83641908f65c3a74d80409d6b1b62ccfaab220b9ea70b80df5a2e0549 446B / 446B 0.4s
=> => extracting sha256:dd71dde834b5c203d162902e6b8994cb2309ae049a0eabc4efea161b2b5a3d0e 2.1s
=> => extracting sha256:1e5a4c89cee5c0826c540ab06d4b6b491c96eda01837f430bd47f0d26702d6e3 0.1s
=> => extracting sha256:25ff2da83641908f65c3a74d80409d6b1b62ccfaab220b9ea70b80df5a2e0549 0.0s
=> [internal] load build context                                0.2s
=> => transferring context: 415.93kB                             0.1s
=> [2/4] WORKDIR /app                                          0.3s
=> [3/4] COPY . .                                              0.0s
=> [4/4] RUN npm install                                       4.2s
=> exporting to image                                          0.3s
=> => exporting layers                                           0.3s
=> => writing image sha256:f038b3bc2f0eb27b198ff7119f2acefbef832c832b2a635120fe5ebf1153bf44 0.0s
=> => naming to docker.io/library/todo-list-nodejs             0.0s
[ec2-user@ip-172-31-38-230 Todo-List-nodejs]$
```

## Login to GHCR

```
[ec2-user@ip-172-31-38-230 Todo-List-nodejs]$ docker login ghcr.io -u "diaaqqassem"
Password:
WARNING! Your password will be stored unencrypted in /home/ec2-user/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store
```

Login Succeeded

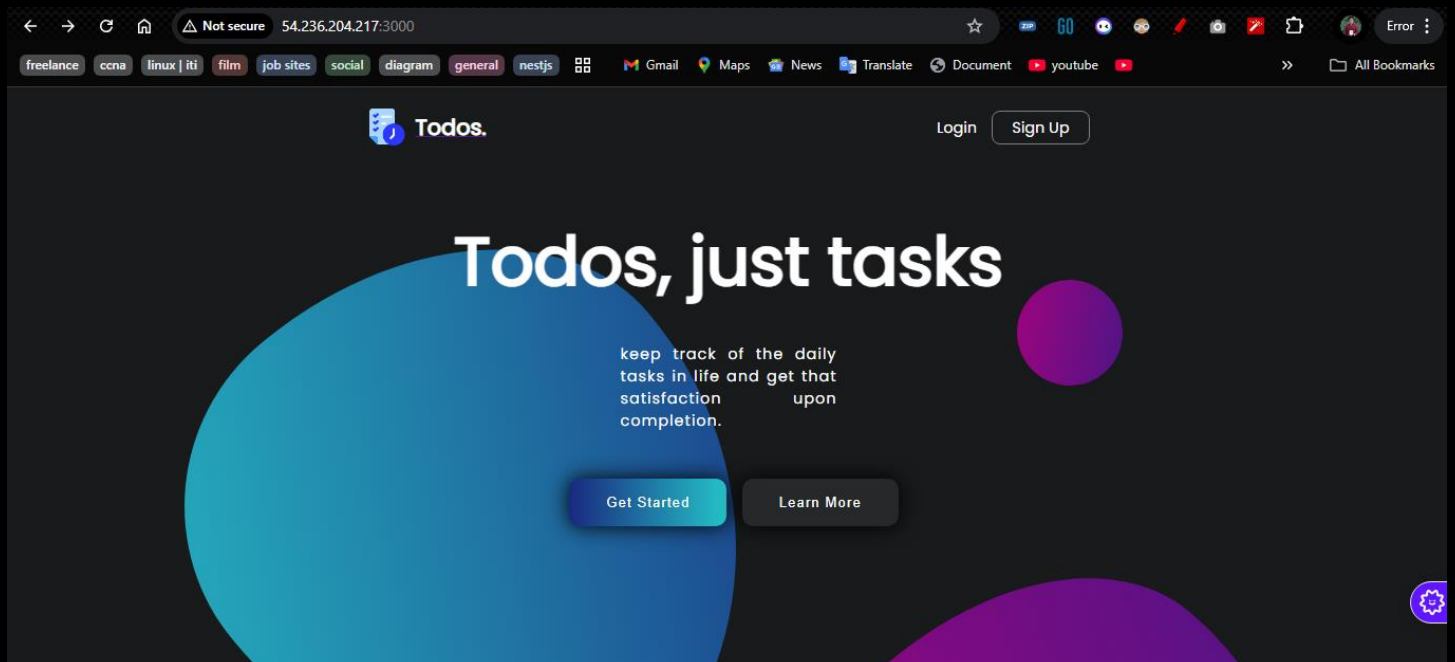
```
[ec2-user@ip-172-31-38-230 Todo-List-nodejs]$
```

## Test

```
[ec2-user@ip-172-31-38-230 Todo-List-nodejs]$ docker run -p 3000:4000 --name test-todo todo-list-nodejs
```

```
> todo-list-nodejs@1.0.0 start
> nodemon index.js nodemon
```

```
[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node index.js nodemon`
27 Jul 2025
Router loaded
Sun, 27 Jul 2025 19:14:25 GMT body-parser deprecated undefined extended: provide extended option at
index.js:25:17
Yupp! Server is running on port 4000
```



```
test> show databases;
admin          40.00 KiB
chat-app       92.00 KiB
config         48.00 KiB
local          72.00 KiB
todolistDb     24.00 KiB
test> use todolistDb;
switched to db todolistDb
todolistDb> show tables;
dashboards
logins
registers
todolistDb> 
```

# GitHub Action

app-ci.yml X

```
1  name: Build and Push Docker Image
2  on:
3    push:
4      branches:
5        - master
6  jobs:
7    build-and-push:
8      runs-on: ubuntu-latest
9
10     steps:
11       - name: Checkout code
12         uses: actions/checkout@v3
13
14       - name: Set up Docker
15         uses: docker/setup-buildx-action@v2
16
17       - name: Log in to GitHub Container Registry
18         uses: docker/login-action@v2
19         with:
20           registry: ghcr.io
21           username: ${ github.actor }
22           password: ${ secrets.GITHUB_TOKEN }
23
24       - name: Build and push Docker image
25         uses: docker/build-push-action@v4
26         with:
27           context: .
28           push: true
29           tags: ghcr.io/${ github.repository_owner }/todo-list-nodejs:latest
--
```

diaaqqassem / Todo-List-nodejs

Type to search

<> Code

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

Actions

New workflow

All workflows

Build and Push Docker Image

Management

Caches

Attestations

Runners

Usage metrics

Performance metrics

All workflows

Showing runs from all workflows

Filter workflow runs

1 workflow run

EventStatusBranchActor

add github action

Build and Push Docker Image #1: Commit acb642 pushed by diaaqqassem

master

now

In progress

...

diaaqqassem / Todo-List-nodejs

Type to search

<> Code

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

Build and Push Docker Image

add github action #1

Re-run all jobs

Summary

Jobs

build-and-push

Run details

Usage

Workflow file

Triggered via push now

diaaqqassem pushed -> acb642 master

Status

Success

Total duration

21s

Artifacts

-

app-ci.yml

on: push

build-and-push

18s

```
"image.name": "ghcr.io/diaaqqassem/todo-list-nodejs:latest"
```

```
[ec2-user@ip-172-31-38-230 Todo-List-nodejs]$ docker pull ghcr.io/diaaqqassem/todo-list-nodejs:latest
latest: Pulling from diaaqqassem/todo-list-nodejs
f18232174bc9: Already exists
dd71dde834b5: Already exists
1e5a4c89cee5: Already exists
25ff2da83641: Already exists
65d67a9b9b49: Pull complete
e1332b61c611: Pull complete
6f0f13240119: Pull complete
Digest: sha256:14481570791835c70e8a19c27bf2b19736508626696eb082640ffbab8854d541
Status: Downloaded newer image for ghcr.io/diaaqqassem/todo-list-nodejs:latest
ghcr.io/diaaqqassem/todo-list-nodejs:latest
[ec2-user@ip-172-31-38-230 Todo-List-nodejs]$
```

```
[ec2-user@ip-172-31-38-230 Todo-List-nodejs]$ docker run -p 3000:4000 --name test-todo ghcr.io/diaaqqassem/todo-list-nodejs:latest
```

```
> todo-list-nodejs@1.0.0 start
> nodemon index.js nodemon
```

```
[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node index.js nodemon`
```

```
27 Jul 2025
```

```
Router loaded
```

```
Sun, 27 Jul 2025 20:02:26 GMT body-parser deprecated undefined extended: provide extended option at
index.js:25:17
```

```
Yupp! Server is running on port 4000
connected to database
```

## Part 2

```
[defaults]
inventory=./hosts
remote_user=ec2-user
ask_pass=false

[privilege_escalation]
become=true
become_ask_pass=false
become_method=sudo
~
~
~
~
```

```
[diaa@ITI Fortstak]$ vi ansible.cfg
[diaa@ITI Fortstak]$ ansible all -m ping
[WARNING]: Invalid characters were found in group names but not replaced, use
[WARNING]: Platform linux on host 54.236.204.217 is using the discovered Python
/usr/bin/python3.9, but future installation of another Python interpreter could
of that path. See https://docs.ansible.com/ansible-core/2.14/reference_appendices/interpreter_discovery.html for more information
54.236.204.217 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.9"
  },
  "changed": false,
  "ping": "pong"
}
[diaa@ITI Fortstak]$
```



```

- name: Install Docker on EC2
  hosts: ToDo-Server
  become: yes
  tasks:
    - name: Install Docker
      yum:
        name: docker
        state: present

    - name: Start and enable Docker
      service:
        name: docker
        state: started
        enabled: yes

    - name: Add current user to docker group
      user:
        name: "{{ ansible_user }}"
        groups: docker
        append: yes

```

playbook..

```

[diaa@ITI Fortstak]$ ansible-playbook playbook.yml
[WARNING]: Invalid characters were found in group names but not replaced, use -vvvv to see details

PLAY [Install Docker on EC2] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host 54.236.204.217 is using the discovered Python interpreter at
/usr/bin/python3.9, but future installation of another Python interpreter could change the meaning
of that path. See https://docs.ansible.com/ansible-
core/2.14/reference_appendices/interpreter_discovery.html for more information.
ok: [54.236.204.217]

TASK [Install Docker] *****
ok: [54.236.204.217]

TASK [Start and enable Docker] *****
ok: [54.236.204.217]

TASK [Add current user to docker group] *****
ok: [54.236.204.217]

PLAY RECAP *****
54.236.204.217      : ok=4    changed=0    unreachable=0    failed=0    skipped=0    rescued=
0    ignored=0

[diaa@ITI Fortstak]$

```

### Part 3 (40 points):

- On the VM, use docker compose to run the application. Make sure you configure the proper health checks.
- You need to continuously check for changes in the image on the docker registry repo. If a change is detected, the new image should be pulled.
- For the auto update part, you can use any tool of your choice. You would need to justify your choice.

### For Part 3

docker-compose.yml - The Compose specification establishes a standard for the definition of multi-container platform-a

```
version: "3.9"
services:
  myapp:
    image: ghcr.io/diaaqaassem/todo-list-nodejs:latest
    ports:
      - "3000:4000"
    restart: always
    healthcheck:
      test: ["CMD", "curl", "-f", "http://localhost:3000/"]
      interval: 30s
      timeout: 10s
      retries: 3
```

```
[ec2-user@ip-172-31-38-230 docker-compose]$ docker compose up -d
WARN[0000] /home/ec2-user/Fortstack/ToDo-List-nodejs/docker-compose/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 2/2
  ✓ Network docker-compose_default    Created                                0.1s
  ✓ Container docker-compose-myapp-1  Started                             0.4s
[ec2-user@ip-172-31-38-230 docker-compose]$ docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED
STATUS        PORTS                               NAMES
e9c18f30d931  ghcr.io/diaaqaassem/todo-list-nodejs:latest  "docker-entrypoint.s..." 5 seconds ago
Up 4 seconds (health: starting)  0.0.0.0:3000->4000/tcp, :::3000->4000/tcp  docker-compose-myapp-1
[ec2-user@ip-172-31-38-230 docker-compose]$ curl localhost:3000
<html>

<head>
  <title>
    Todos
  </title>
  <link rel="stylesheet" href="/css/home.css">
  <link rel="icon" href="https://cdn-icons-png.flaticon.com/512/7692/7692809.png">
  ..
*5 * * * * cd "/home/ec2-user/Fortstack/ToDo-List-nodejs/docker-compose/" && docker compose pull &&
  docker compose up -d
~
~
```

### Bonus

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: myapp
5  spec:
6    replicas: 2
7    selector:
8      matchLabels:
9        app: myapp
10   template:
11     metadata:
12       labels:
13         app: myapp
14     spec:
15       containers:
16         - name: myapp
17           image: ghcr.io/diaaqaassem/todo-list-nodejs:latest
18           ports:
19             - containerPort: 4000
20           livenessProbe:
21             httpGet:
22               path: /
23               port: 4000
24             initialDelaySeconds: 10
25             periodSeconds: 30
26           readinessProbe:
27             httpGet:
28               path: /
29               port: 4000
30             initialDelaySeconds: 5
31             periodSeconds: 15
32
```

```
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: myapp-service
5  spec:
6    selector:
7      app: myapp
8    ports:
9      - protocol: TCP
10        port: 80
11        nodePort: 30000
12        targetPort: 4000
13    type: NodePort
14
```

```
[ec2-user@ip-172-31-38-230 k8s]$ kubectl apply -f deployment.yml
deployment.apps/myapp created
[ec2-user@ip-172-31-38-230 k8s]$ kubectl get deploy
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
myapp     0/2     2            0           8s
[ec2-user@ip-172-31-38-230 k8s]$ kubectl get deploy
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
myapp     0/2     2            0           15s
[ec2-user@ip-172-31-38-230 k8s]$ kubectl get po
NAME                                READY   STATUS    RESTARTS   AGE
myapp-7d4886d448-4fswb             0/1     Running   0           20s
myapp-7d4886d448-z5dxsf            0/1     Running   0           20s
[ec2-user@ip-172-31-38-230 k8s]$ kubectl get deploy
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
myapp     2/2     2            2           36s
[ec2-user@ip-172-31-38-230 k8s]$
```

```
[ec2-user@ip-172-31-38-230 k8s]$ kubectl get svc
NAME            TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
kubernetes      ClusterIP   10.96.0.1    <none>        443/TCP          15m
myapp-service   NodePort    10.106.52.35 <none>        80:30000/TCP     6m9s
[ec2-user@ip-172-31-38-230 k8s]$ curl 10.106.52.35
<html>

<head>
  <title>
    Todos
  </title>
  <link rel="stylesheet" href="/css/home.css">
  <link rel="icon" href="https://cdn-icons-png.flaticon.com/512/7692/7692809.png">
</head>

<body>
  <div class="nav">
    <div class="nav-logo-section">
      <a href=""></a>
      <a href=""><span class="nav-logo-name">Todos.</span></a>
    </div>
```