

# Week 5: Server Side Foundations with Docker & DevOps Basics

## New Objective:

Help interns understand **how production servers work**, using Docker to simulate deployment environments locally.

## Week-5 Topics

### DAY 1 — Docker Fundamentals + Linux Internals

- Images, containers, volumes, networks
- Basic container OS operations:
  - Users, permissions, logs, processes inside containers
- Dockerfile basics

#### Exercise

- Container running Node app
- Enter container (like SSH):

```
docker exec -it <container> /bin/sh
```

Explore:

- `ls`, `ps`, `top`, disk usage, logs
- Document structure

#### Output

- Dockerfile
- `linux-in-container.md`

---

## DAY 2 — Docker Compose + Multi-Container Apps

- Docker networking
- Volumes for persistent storage
- Compose for multiple services

### Exercise

Deploy:

client (React) + server (Node) + MongoDB

using one command:

```
docker compose up -d
```

Ensure:

- ✓ Server connects Mongo via container networking
- ✓ Logs exposed correctly
- ✓ Volumes mapped

### Output

- `docker-compose.yml`
- `service-architecture.md`

---

## DAY 3 — NGINX Reverse Proxy + Load Balancing

- NGINX inside Docker
- Routing to internal containers
- Multiple backend replicas (load balancing simulation)

Exercise :

- Run **two instances** of backend
- Use NGINX container as reverse proxy:

/api → backend-service:3000

Note : Enable round-robin load balancing

### Output

- `nginx.conf`
  - `reverse-proxy-readme.md`
- 

## DAY 4 — SSL + Self-Signed + mkcert + HTTPS

- SSL/TLS fundamentals
- Setting certs inside container
- HTTPS termination at NGINX

### Exercise

- Generate self-signed cert using mkcert
- Force redirect HTTP → HTTPS
- Confirm lock icon in browser using local domain

### Output

- Certificates + `ssl-setup.md`
  - HTTPS working screenshot
- 

## DAY 5 — CI-Style Deployment Automation + Capstone

### Tools:

- Docker volumes
- Compose profiles
- `.env` config
- Log rotation
- Health checks

## Capstone

Deploy a **full-stack app** in Docker with reverse proxy + HTTPS

Checklist:

- ✓ `docker-compose.prod.yml`
- ✓ Secrets in `.env` (not committed)
- ✓ Healthcheck in compose
- ✓ Container restart policy
- ✓ Deployment script:

## Output

- Fully working application stack
- `production-guide.md`
- Demo recording (optional)