

# LAB 3

## Implementation of Deployment Tools

### Objective:

The goal of this lab was to learn about the importance of deployment tools, focusing mainly on Docker for packaging and deploying applications. We also explored alternative hosting platforms like Render, Vercel, and GitHub Pages.

### Tools Used:

- **Docker** — for containerizing applications
- **Node.js** — to build a basic web app
- **Render** — for simple app deployment
- **Vercel** — for hosting frontend frameworks and serverless projects
- **GitHub Pages** — for hosting static websites
- **VS Code / Terminal** — for development and running commands

### Steps Followed:

#### A. Create a Simple Node.js Web App

##### 1. Initialize the project:

```
mkdir docker-lab && cd docker-lab  
npm init -y  
npm install express
```

##### 2. Create server.js

```
const express = require('express');  
const app = express();  
const PORT = process.env.PORT || 3000;  
  
app.get('/', (req, res) => res.send('Hello from Docker!'));  
  
app.listen(PORT, () => console.log(`Server running on port ${PORT}`));
```

## B. Create a Dockerfile

```
# Use an official Node.js image
FROM node:18

# Set working directory
WORKDIR /app

# Copy app files
COPY package*.json ./
RUN npm install
COPY . .

# Expose port and run app
EXPOSE 3000
CMD ["node", "server.js"]
```

## C. Build and Run Docker Image

```
docker build -t docker-lab .
docker run -p 3000:3000 docker-lab
```

Visit <http://localhost:3000> to see the app running.

## D. Overview of Other Deployment Tools

### 1. Render

- Connect your GitHub repository
- Select Web Service or Static Site
- Configure build and start commands
- Deploy with one click

### 2. Vercel

- Import projects from GitHub
- Auto-deploy on every push
- Great for frontend frameworks like React, Next.js

### 3. GitHub Pages

- Push your static site code to GitHub
- Go to repository Settings → Pages
- Choose branch and folder (e.g. `main` and `/root`)
- GitHub hosts it for free

## Conclusion

This lab offered practical experience in deploying applications using Docker, ensuring consistent environments through containerization. It also introduced modern, beginner-friendly hosting services like Render, Vercel, and GitHub Pages that simplify deployment for both dynamic and static web projects.