# Deploying React + Node.js Full Stack App to Render with GitHub CI/CD

This guide walks you through deploying a full-stack React (frontend) + Node.js (backend) application to Render, using GitHub for source control and CI/CD.

## Step 1: Push Code to GitHub

Ensure your project is in a Git repository and pushed to GitHub. Example project structure:

my-app/  
├── client/ # React App  
├── server/ # Node.js/Express Backend  
├── .git/ # Git repository

Use the following commands if not already pushed:

git init  
git add .  
git commit -m "Initial commit"  
git remote add origin https://github.com/your-username/your-repo.git  
git push -u origin main

## Step 2: Deploy Backend (Node.js) to Render

1. Go to https://render.com  
2. Click “New +” > Web Service  
3. Connect your GitHub repository  
4. Select the server folder (Node backend)  
5. Fill in:  
 - Environment: Node  
 - Build Command: npm install  
 - Start Command: node index.js (or your entry file)  
6. Add required environment variables (e.g., DB\_URI, PORT)  
7. Click Create Web Service  
  
Render automatically redeploys the service on new GitHub pushes.

## Step 3: Deploy Frontend (React) to Render

1. Click “New +” > Static Site  
2. Connect to the same GitHub repository  
3. Select the client folder (React frontend)  
4. Set:  
 - Build Command: npm install && npm run build  
 - Publish directory: build  
5. Add any required build-time environment variables  
6. Click Create Static Site

## CI/CD Integration

Render supports automatic deployment from GitHub. Each time you push to the main branch, Render will redeploy both your backend and frontend services automatically.

## Optional: Proxy React to Backend

In your React package.json, you can add a proxy setting to avoid CORS issues during development:

"proxy": "https://your-backend-service.onrender.com"

## Deployment Summary

✅ Frontend (React): https://your-react-app.onrender.com  
✅ Backend (Node): https://your-backend-service.onrender.com