



Complete GitHub Deployment Guide

Table of Contents

1. Prerequisites
2. Step-by-Step GitHub Deployment
3. Environment Variables Setup
4. Deployment Options
5. Troubleshooting

Prerequisites

Before you begin, ensure you have:

- A GitHub account ([Sign up here](https://github.com/signup) (<https://github.com/signup>))
- Git installed on your machine ([Download here](https://git-scm.com/downloads) (<https://git-scm.com/downloads>))
- Access to your database credentials
- AWS S3 credentials (for image uploads)

Step-by-Step GitHub Deployment

Step 1: Create a GitHub Repository

1. Go to [GitHub](https://github.com) (<https://github.com>) and log in
2. Click the "+" icon in the top-right corner → “**New repository**”
3. Fill in the details:
 - **Repository name:** radians-automation-website
 - **Description:** “Industrial automation company website with admin CMS”
 - **Visibility:** Select **Private** (recommended for business sites)
 - **DO NOT** check any initialization options (README, .gitignore, license)
4. Click “**Create repository**”
5. **Keep this page open** - you’ll need the repository URL

Step 2: Initialize Git in Your Project

Open your terminal and navigate to the project directory:

```
cd /home/ubuntu/radians_automation_website/nextjs_space
```

Initialize Git repository:

```
git init
```

Configure your Git identity (if not already done):

```
git config --global user.name "Your Name"  
git config --global user.email "your.email@example.com"
```

Step 3: Add Files to Git

Add all files to staging:

```
git add .
```

Verify what will be committed:

```
git status
```

You should see all your project files listed in green. The `.env` file should NOT appear (it's excluded by `.gitignore`).

Step 4: Create Your First Commit

```
git commit -m "Initial commit: Radians Automation website with admin CMS"
```

Step 5: Connect to GitHub Repository

Copy the repository URL from your GitHub page (it looks like:

https://github.com/YOUR_USERNAME/radians-automation-website.git)

Add the remote repository:

```
git remote add origin https://github.com/YOUR_USERNAME/radians-automation-website.git
```

Verify the remote was added:

```
git remote -v
```

Step 6: Push Your Code to GitHub

Push your code to the main branch:

```
git branch -M main  
git push -u origin main
```

If prompted for credentials:

- Username: Your GitHub username
- Password: Use a **Personal Access Token** (not your GitHub password)

To create a Personal Access Token:

1. Go to GitHub → Settings → Developer settings → Personal access tokens → Tokens (classic)
 2. Click “Generate new token (classic)”
 3. Give it a name: “Radians Automation Deploy”
 4. Select scopes: Check **repo** (all permissions)
 5. Click “Generate token”
 6. **Copy the token immediately** (you won’t see it again!)
 7. Use this token as your password when pushing
-

Step 7: Verify Your Repository

1. Go to your GitHub repository page
 2. Refresh the page
 3. You should see all your files uploaded! ✓
-

Environment Variables Setup

⚠️ IMPORTANT: Your `.env` file is NOT pushed to GitHub (for security reasons).

For Local Development:

Create a `.env` file in the `nextjs_space` directory:

```
cp .env.example .env
```

Then edit `.env` with your actual credentials:

```
DATABASE_URL="postgresql://your-db-connection-string"
NEXTAUTH_SECRET="your-secret-key"
NEXTAUTH_URL="http://localhost:3000"
AWS_BUCKET_NAME="your-bucket-name"
AWS_FOLDER_PREFIX="radians-automation/"
AWS_ACCESS_KEY_ID="your-access-key"
AWS_SECRET_ACCESS_KEY="your-secret-key"
AWS_REGION="us-east-1"
```

For Production Deployment:

You'll need to add these environment variables to your hosting platform (see Deployment Options below).

Deployment Options

Option 1: Deploy to Vercel (Recommended)

Vercel is the easiest way to deploy Next.js applications:

1. **Go to Vercel (<https://vercel.com>) and sign up/login**
2. Click “Add New...” → “Project”
3. **Import your GitHub repository:**
 - Select “Import Git Repository”
 - Choose your `radians-automation-website` repository
 - Click “Import”
4. **Configure your project:**
 - **Framework Preset:** Next.js (should be auto-detected)
 - **Root Directory:** `nextjs_space`
 - **Build Command:** `yarn build` (default)
 - **Output Directory:** `.next` (default)
5. **Add Environment Variables:**
 - Click “Environment Variables”
 - Add each variable from your `.env` file:
 - `DATABASE_URL` = `your-database-url`
 - `NEXTAUTH_SECRET` = `your-secret`
 - `NEXTAUTH_URL` = `https://your-domain.vercel.app`
 - `AWS_BUCKET_NAME` = `your-bucket`
 - `AWS_FOLDER_PREFIX` = `radians-automation/`
 - `AWS_ACCESS_KEY_ID` = `your-key`
 - `AWS_SECRET_ACCESS_KEY` = `your-secret`
 - `AWS_REGION` = `us-east-1`
6. **Deploy:**
 - Click “Deploy”
 - Wait 2-3 minutes for deployment to complete
 - Your site will be live at `https://your-project.vercel.app`

Custom Domain Setup on Vercel:

1. Go to your project settings → “Domains”
 2. Add your custom domain
 3. Follow Vercel’s instructions to update your DNS records
-

Option 2: Deploy to Netlify

1. **Go to Netlify (<https://netlify.com>) and sign up/login**
2. Click “Add new site” → “Import an existing project”
3. **Connect to GitHub:**
 - Select GitHub
 - Choose your repository
4. **Configure build settings:**
 - **Base directory:** `nextjs_space`

- **Build command:** yarn build

- **Publish directory:** .next

5. Add Environment Variables:

- Go to Site settings → Environment variables
- Add all your environment variables

6. Deploy site

Option 3: Deploy to Your Own Server (VPS/Dedicated)

For deployment to a VPS (DigitalOcean, AWS EC2, Linode, etc.):

1. Set up your server:

```
# Install Node.js 18+
curl -fsSL https://deb.nodesource.com/setup_18.x | sudo -E bash -
sudo apt-get install -y nodejs

# Install Yarn
npm install -g yarn

# Install PM2 (process manager)
npm install -g pm2
```

2. Clone your repository on the server:

```
git clone https://github.com/YOUR_USERNAME/radians-automation-website.git
cd radians-automation-website/nextjs_space
```

3. Install dependencies:

```
yarn install
```

4. Set up environment variables:

```
nano .env
# Paste your environment variables
# Save with Ctrl+X, then Y, then Enter
```

5. Set up database:

```
yarn prisma generate
yarn prisma migrate deploy
yarn prisma db seed
```

6. Build the application:

```
yarn build
```

7. Start with PM2:

```
pm2 start yarn --name "radians-automation" -- start
pm2 save
pm2 startup
```

8. Set up Nginx as reverse proxy:

```
sudo apt install nginx
sudo nano /etc/nginx/sites-available/radians-automation
```

Add this configuration:

```
server {
    listen 80;
    server_name your-domain.com;

    location / {
        proxy_pass http://localhost:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
    }
}
```

Enable the site:

```
sudo ln -s /etc/nginx/sites-available/radians-automation /etc/nginx/sites-enabled/
sudo nginx -t
sudo systemctl restart nginx
```

9. Set up SSL with Let's Encrypt:

```
sudo apt install certbot python3-certbot-nginx
sudo certbot --nginx -d your-domain.com
```

Updating Your Deployed Application

Push Updates to GitHub:

```
# Make your changes, then:
git add .
git commit -m "Description of your changes"
git push origin main
```

Auto-Deploy:

- **Vercel/Netlify:** Automatically deploys when you push to GitHub! 
- **Own Server:** SSH into your server and run:

```
cd /path/to/radians-automation-website/nextjs_space
git pull origin main
yarn install
yarn build
pm2 restart radians-automation
```

Troubleshooting

Issue: “Permission denied (publickey)”

Solution: Set up SSH key or use HTTPS with Personal Access Token

Issue: “.env variables not working in production”

Solution: Make sure you added ALL environment variables in your hosting platform’s dashboard

Issue: “Database connection failed”

Solution:

- Verify your DATABASE_URL is correct
- Ensure your database allows connections from your deployment platform’s IP
- For Vercel: Add your Vercel IPs to database whitelist

Issue: “Images not loading”

Solution:

- Verify AWS credentials are correct
- Check S3 bucket permissions (should allow public read)
- Ensure AWS_REGION matches your bucket region

Issue: “NextAuth error”

Solution:

- Make sure NEXTAUTH_URL is set to your production domain
- Verify NEXTAUTH_SECRET is set (minimum 32 characters)

Additional Resources

- [Next.js Documentation](https://nextjs.org/docs) (<https://nextjs.org/docs>)
- [Vercel Deployment Guide](https://vercel.com/docs) (<https://vercel.com/docs>)
- [GitHub Guides](https://guides.github.com) (<https://guides.github.com>)
- [PM2 Documentation](https://pm2.keymetrics.io/docs/usage/quick-start/) (<https://pm2.keymetrics.io/docs/usage/quick-start/>)

Need Help?

If you encounter any issues:

1. Check the error messages carefully
2. Review the troubleshooting section above

3. Search for the error on Stack Overflow
 4. Contact your development team
-

 **Congratulations!** Your Radians Automation website is now deployed and ready for the world!