

DAY 6

DEPLOYMENT STAGING PROCESS DOCUMENTATION

This document outlines the detailed steps for deploying a project to a staging environment. The process ensures error-free deployment and smooth integration of the application.

Step 1: Push Code to GitHub

The first step involves pushing the latest code changes to the GitHub repository. This ensures version control and accessibility of the codebase for deployment. Use the following command to push the code:

Add changes to the staging area

git add .

git commit -m "finaly complete"00

git push origin main

Step 2: Build Creation in Development Environment

After pushing the code to GitHub, create a build in the development environment. This step validates the integrity of the application and ensures no errors occur during the build process. Run the following command to create the build:

npm install

npm run build

```
✓ Generating static pages (15/15)
✓ Collecting build traces
✓ Finalizing page optimization

Route (app)      Size      First Load JS
┌───┴──────────┴───┐
├─ o /            2.29 kB      151 kB
├─ o /_not-found  905 B        101 kB
├─ o /about-us    177 B        106 kB
├─ o /Cart        3.12 kB      152 kB
├─ o /ceramics    1.95 kB      151 kB
├─ o /chairs      1.95 kB      151 kB
├─ o /crockery    1.95 kB      151 kB
├─ o /cutlery     1.95 kB      151 kB
├─ o /Planpots    1.95 kB      151 kB
├─ o /productlisting 2.07 kB      160 kB
├─ f /productlisting/[id] 1.81 kB      142 kB
├─ o /studio/[[...tool]] 1.28 MB      1.42 MB
├─ o /table       1.95 kB      151 kB
├─ o /tableware   1.95 kB      151 kB
├─ + First Load JS shared by all 100 kB
├─   └─ chunks/1517-841e40f43c47304e.js 45.7 kB
├─   └─ chunks/4bd1b696-25c326edd8443693.js 52.5 kB
├─   └─ other shared chunks (total) 2.29 kB
```

Verify the build locally to confirm that the application functions as expected.

Step 3: Add Environment Variables

To ensure proper configuration, add all required environment variables. These variables are critical for connecting to external services, databases, or APIs.

Steps to add environment variables:

7 Data and Output Settings

Environment Variables

| Key | Value | |
|-------------------------------|----------------------------------|---|
| apiVersion | 2025-01-19 | — |
| NEXT_PUBLIC_SANITY_PROJECT_ID | dy7s2c62 | — |
| NEXT_PUBLIC_SANITY_DATASET | production | — |
| SANITY_API_TOKEN | sk0TRp4e0ijBpJu8opDcSO95DGLjBYqM | — |

+ Add More

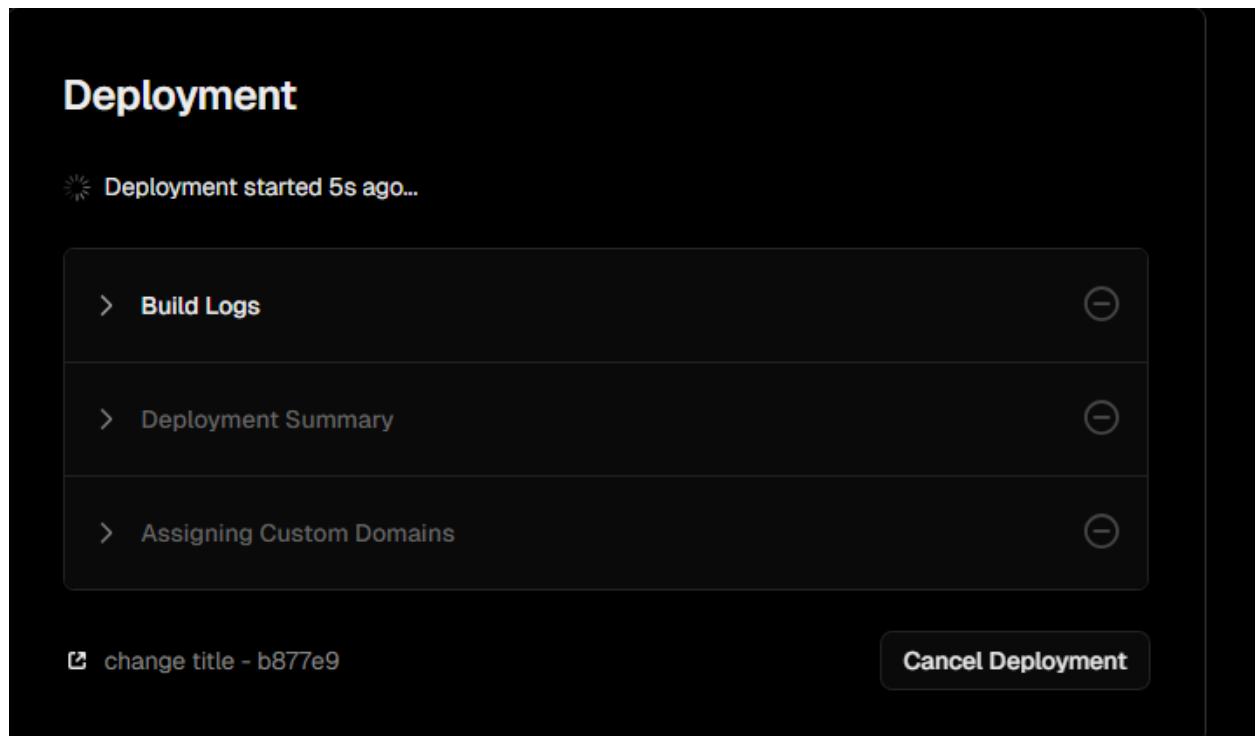
Tip: Paste an .env above to populate the form. [Learn more](#)

Deploy

Step 4: Deployment

Once the build and environment variables are ready, deploy the application to the staging environment. This can be done using a platform like Vercel, Netlify, or any hosting provider of choice.

Deployment Steps (Example: Vercel)



Step 5: Create Webhook

After successful deployment, create a webhook to automate tasks such as notifying external systems about changes or updates.

Webhook Creation Process:

1. Navigate to your project's webhook settings in the hosting provider or CMS (e.g., Sanity,).
 2. Configure the webhook with the following details:
 - **Name:** Staging Deployment Webhook
 - **URL:** `https://your-domain.com/webhooks/staging-update`
 - **Triggers:** Enable triggers for actions like create, update, or delete.
 - **Secret:** Add a secure secret key to verify incoming requests.
 3. Save the webhook configuration.
-