

# Vercel Environment Variables Configuration

---

This document outlines all environment variables required for deploying Studio 2 to Vercel.

## Required Environment Variables

---

### 1. Database Configuration

#### `DATABASE_URL` (REQUIRED)

- **Description:** PostgreSQL database connection string
- **Format:** `postgresql://USER:PASSWORD@HOST:PORT/DATABASE?connect_timeout=15`
- **Example:** `postgresql://user:pass@host.db.vercel.app:5432/dbname?connect_timeout=15`
- **Where to Set:** Vercel Dashboard → Project Settings → Environment Variables
- **Important Notes:**
  - You can use **Vercel Postgres** (recommended for easy setup)
  - Or use an external PostgreSQL provider (Supabase, Railway, Neon, etc.)
  - Ensure `connect_timeout=15` is included for better reliability

### 2. NextAuth Configuration

#### `NEXTAUTH_SECRET` (REQUIRED)

- **Description:** Secret key for NextAuth.js session encryption
- **How to Generate:**

```
bash
openssl rand -base64 32
```

Or use any random string generator
- **Example:** `JiwMZ945W995hca62G5VYmrLaKKqsvf0`
- **Where to Set:** Vercel Dashboard → Project Settings → Environment Variables
- **Security:** NEVER commit this to git or share publicly

#### `NEXTAUTH_URL` (REQUIRED)

- **Description:** Base URL of your deployed application
- **Format:** `https://your-app-name.vercel.app`
- **Example:** `https://studio2-app.vercel.app`
- **Where to Set:** Vercel Dashboard → Project Settings → Environment Variables
- **Important:** Update this after your first deployment with the actual Vercel URL

### 3. AWS Configuration (OPTIONAL)

These are only needed if you're using AWS S3 for file storage:

#### `AWS_PROFILE` (Optional)

- **Description:** AWS profile name
- **Default:** `hosted_storage`

#### `AWS_REGION` (Optional)

- **Description:** AWS region for S3 bucket
- **Example:** `us-west-2`

**AWS\_BUCKET\_NAME (Optional)**

- **Description:** S3 bucket name for file uploads
- **Example:** `your-studio2-bucket`

**AWS\_FOLDER\_PREFIX (Optional)**

- **Description:** Folder prefix for organizing files in S3
- **Example:** `uploads/`

**AWS\_ACCESS\_KEY\_ID (Optional)**

- **Description:** AWS access key for S3 operations
- **Security:** Keep this secret

**AWS\_SECRET\_ACCESS\_KEY (Optional)**

- **Description:** AWS secret access key for S3 operations
- **Security:** Keep this secret

---

## Setting Environment Variables in Vercel

### Step 1: Navigate to Environment Variables

1. Go to [Vercel Dashboard](https://vercel.com/dashboard) (<https://vercel.com/dashboard>)
2. Select your project
3. Click on **Settings** tab
4. Click on **Environment Variables** in the sidebar

### Step 2: Add Each Variable

1. Enter the **Key** (variable name, e.g., `DATABASE_URL` )
2. Enter the **Value** (the actual value)
3. Select which environments to apply to:
  - ☒ **Production** (for your live site)
  - ☒ **Preview** (for pull request deployments)
  - ☒ **Development** (for local development with `vercel dev` )
4. Click **Save**

### Step 3: Redeploy

After adding or updating environment variables, trigger a new deployment:

- Go to **Deployments** tab
- Click on the latest deployment
- Click **Redeploy**

---

## Database Setup Options

### Option 1: Vercel Postgres (Recommended for Beginners)

1. Go to your Vercel project
2. Click on **Storage** tab
3. Click **Create Database → Postgres**

4. Follow the setup wizard
5. Vercel will automatically add `DATABASE_URL` to your environment variables
6. Run migrations: `npx prisma migrate deploy`

## Option 2: External PostgreSQL Provider

### Supabase

1. Create account at [supabase.com](https://supabase.com) (<https://supabase.com>)
2. Create a new project
3. Get connection string from Project Settings → Database
4. Use the “URI” connection string format
5. Add to Vercel as `DATABASE_URL`

### Railway

1. Create account at [railway.app](https://railway.app) (<https://railway.app>)
2. Create new project → Add PostgreSQL
3. Copy the “Postgres Connection URL”
4. Add to Vercel as `DATABASE_URL`

### Neon

1. Create account at [neon.tech](https://neon.tech) (<https://neon.tech>)
2. Create a new project
3. Copy the connection string
4. Add to Vercel as `DATABASE_URL`

---

## Running Database Migrations on Vercel

---

After setting up your database, you need to run Prisma migrations:

### Method 1: Automatic (via build script)

The `vercel.json` includes `npx prisma generate` in the build command, which handles schema generation.

To run migrations, add this to your `package.json` :

```
{
  "scripts": {
    "postinstall": "npx prisma generate",
    "vercel-build": "npx prisma migrate deploy && next build"
  }
}
```

## Method 2: Manual (via Vercel CLI)

```
# Install Vercel CLI
npm i -g vercel

# Login
vercel login

# Link to your project
vercel link

# Run migrations
vercel env pull .env.local
npx prisma migrate deploy
```

---

## Environment Variables Checklist

Before deploying, ensure you have set:

- [ ] `DATABASE_URL` - PostgreSQL connection string
- [ ] `NEXTAUTH_SECRET` - Generated secret for authentication
- [ ] `NEXTAUTH_URL` - Your Vercel deployment URL
- [ ] AWS variables (if using S3 storage)
- [ ] Any custom API keys or secrets your app uses

---

## Security Best Practices

1. **Never commit** `.env` files to git - They're in `.gitignore` for a reason
2. **Use** `.env.example` - Document required variables without exposing secrets
3. **Rotate secrets regularly** - Especially after team member changes
4. **Use different secrets** - Different values for production vs. preview environments
5. **Limit access** - Only give team members access to production secrets if necessary

---

## Troubleshooting

### “Invalid `prisma.client` invocation”

- Ensure `DATABASE_URL` is set correctly
- Check database is accessible from Vercel's IP ranges
- Verify connection string format

### “`NEXTAUTH_URL` is not defined”

- Add `NEXTAUTH_URL` to environment variables
- Must match your actual deployment URL

## Database connection timeout

- Add `?connect_timeout=15` to your connection string
- Check your database provider's connection limits

## Prisma Client not generated

- Ensure `npx prisma generate` runs during build
- Check build logs in Vercel dashboard

---

## Need Help?

- [Vercel Environment Variables Docs](https://vercel.com/docs/concepts/projects/environment-variables) (<https://vercel.com/docs/concepts/projects/environment-variables>)
- [Vercel Postgres Docs](https://vercel.com/docs/storage/vercel-postgres) (<https://vercel.com/docs/storage/vercel-postgres>)
- [NextAuth.js Docs](https://next-auth.js.org/configuration/options) (<https://next-auth.js.org/configuration/options>)
- [Prisma with Vercel Docs](https://www.prisma.io/docs/guides/deployment/deployment-guides/deploying-to-vercel) (<https://www.prisma.io/docs/guides/deployment/deployment-guides/deploying-to-vercel>)