






Studio 2 - Vercel Deployment Summary

What's Been Completed




Your Studio 2 NextJS application has been fully prepared for Vercel deployment via GitHub! Here's what was done:

Configuration Updates





1. Next.js Configuration (`next.config.js`)

-  Removed custom `distDir` (Vercel handles this automatically)
-  Removed `outputFileTracingRoot` (incompatible with Vercel)
-  Removed `output` mode setting (use Vercel's default)
-  Enabled proper ESLint and TypeScript configuration
-  Updated image optimization settings for Vercel




2. Prisma Configuration (`prisma/schema.prisma`)

-  Removed hardcoded `output` path (breaks on Vercel)
-  Updated `binaryTargets` for Vercel's Linux environment
-  Configured for PostgreSQL database




3. Vercel Configuration (`vercel.json`) - NEW

-  Created custom build command with Prisma generation
-  Set API route timeout limits (30 seconds)
-  Configured CORS headers for API routes
-  Set deployment region preference

4. Environment Variables Documentation

-  Created `.env.example` with all required variables
-  Documented each variable in `VERCEL_ENV_VARIABLES.md`
-  Provided database setup options (Vercel Postgres, Supabase, Railway, Neon)

5. Git Configuration

-  Connected to GitHub repository: `https://github.com/Stefanbotes/Studiov2.git`
 -  Committed all changes with descriptive message
 -  Ready to push to GitHub (authentication required)
-

Documentation Created

Main Guides

1. **VERCEL_DEPLOYMENT_GUIDE.md** ★
 - Complete step-by-step deployment instructions

- Database setup options
- Environment variables configuration
- Post-deployment checklist
- Monitoring and maintenance tips

2. **VERCEL_ENV_VARIABLES.md** 🗝️

- Detailed documentation of all environment variables
- How to generate secrets
- Where to set each variable
- Database connection string formats
- AWS S3 configuration (optional)

3. **TROUBLESHOOTING.md** 🔧

- Common deployment issues and solutions
- Build errors and fixes
- Database connection problems
- Authentication issues
- Runtime error debugging

4. **GITHUB_PUSH_INSTRUCTIONS.md** 📦

- How to authenticate with GitHub
- Personal Access Token setup
- SSH key configuration
- GitHub CLI usage
- Push troubleshooting



Files Modified

Updated Files

- `next.config.js` - Vercel-compatible configuration
- `prisma/schema.prisma` - Fixed for Vercel deployment
- `.gitignore` - Already configured correctly

New Files

- `vercel.json` - Vercel deployment configuration
- `.env.example` - Environment variables template
- `VERCEL_DEPLOYMENT_GUIDE.md` - Comprehensive deployment guide
- `VERCEL_ENV_VARIABLES.md` - Environment variables documentation
- `TROUBLESHOOTING.md` - Common issues and solutions
- `GITHUB_PUSH_INSTRUCTIONS.md` - GitHub authentication guide
- `DEPLOYMENT_SUMMARY.md` - This file

Removed Files

- `.yarn/install-state.gz` - Removed (using npm)
- `.yarnrc.yml` - Removed (using npm)
- `yarn.lock` - Removed (using npm)
- `prisma/studio_2.db` - Removed (local SQLite file)

Next Steps - Your Action Required

Step 1: Push to GitHub (Required)

Your code is ready but needs to be pushed to GitHub. You'll need to authenticate.

Choose ONE authentication method:

Option A: Personal Access Token (Recommended)

1. Create token at: <https://github.com/settings/tokens>

2. Select scope: `repo` and `workflow`

3. Run:

```
bash
cd /home/ubuntu/studio_2_app/nextjs_space
git push -u origin master
```

4. Use token as password when prompted

Option B: GitHub CLI

```
gh auth login
cd /home/ubuntu/studio_2_app/nextjs_space
git push -u origin master
```

Option C: SSH Key

1. Generate SSH key: `ssh-keygen -t ed25519`

2. Add to GitHub: <https://github.com/settings/keys>

3. Update remote: `git remote set-url origin git@github.com:Stefanbotes/Studiov2.git`

4. Push: `git push -u origin master`

 **Detailed instructions:** See `GITHUB_PUSH_INSTRUCTIONS.md`

Step 2: Deploy to Vercel

Once code is on GitHub:

1. Import to Vercel

- Go to <https://vercel.com/dashboard>
- Click "Add New..." → "Project"
- Import `Stefanbotes/Studiov2` repository
- **Root Directory:** Set to `nextjs_space` (if app is in subdirectory)
- Deploy (first deploy may fail - expected)

2. Set Up Database

Option A: Vercel Postgres (Easiest)

- In Vercel dashboard → Storage → Create Database → Postgres
- `DATABASE_URL` is added automatically
- Run migrations (see deployment guide)

Option B: External Database

- Use Supabase, Railway, or Neon

- Get connection string
- Add to Vercel environment variables

1. Configure Environment Variables

Go to Settings → Environment Variables and add:

Required:

- `DATABASE_URL` - Your PostgreSQL connection string
- `NEXTAUTH_SECRET` - Generate with: `openssl rand -base64 32`
- `NEXTAUTH_URL` - Your Vercel deployment URL (update after first deploy)

Optional (AWS S3):

- `AWS_PROFILE`
- `AWS_REGION`
- `AWS_BUCKET_NAME`
- `AWS_FOLDER_PREFIX`
- `AWS_ACCESS_KEY_ID`
- `AWS_SECRET_ACCESS_KEY`

1. Redeploy

- Go to Deployments tab
- Click “Redeploy”
- Monitor build logs

2. Run Database Migrations

```
bash
vercel env pull .env.local
npx prisma migrate deploy
```

3. Update NEXTAUTH_URL

- Get your deployment URL from Vercel
- Update `NEXTAUTH_URL` environment variable
- Redeploy one more time

 **Detailed instructions:** See `VERCEL_DEPLOYMENT_GUIDE.md`



Environment Variables Checklist

Before deploying, ensure you have:

- ☐ `DATABASE_URL` - PostgreSQL connection string
- ☐ `NEXTAUTH_SECRET` - Generated secret (32+ characters)
- ☐ `NEXTAUTH_URL` - Your Vercel deployment URL
- ☐ AWS credentials (if using S3 storage)



What to Check After Deployment

1. Application Access

- ☐ Can access homepage

- [] No console errors
- [] Styling loads correctly

2. Authentication

- [] Can register new account
- [] Can login
- [] Session persists across page reloads

3. Database Functionality

- [] Can create client profiles
- [] Can view existing data
- [] Can update records

4. Features

- [] JSON import works
- [] Client profile management
- [] Coaching sessions functionality
- [] Assessment tools



Project Structure

```
studio_2_app/nextjs_space/
├── app/                # Next.js 14 app directory
├── components/         # React components
├── lib/               # Utility functions
├── prisma/            # Database schema and migrations
│   └── schema.prisma  # ✅ Updated for Vercel
├── public/            # Static files
├── .env.example        # ✅ NEW - Environment variables template
├── .gitignore          # Git ignore rules
├── next.config.js      # ✅ Updated for Vercel compatibility
├── package.json        # Dependencies
├── vercel.json         # ✅ NEW - Vercel configuration
├── VERCEL_DEPLOYMENT_GUIDE.md # ✅ NEW - Deployment instructions
├── VERCEL_ENV_VARIABLES.md # ✅ NEW - Environment variables docs
├── TROUBLESHOOTING.md  # ✅ NEW - Common issues and fixes
├── GITHUB_PUSH_INSTRUCTIONS.md # ✅ NEW - GitHub authentication guide
└── DEPLOYMENT_SUMMARY.md # ✅ NEW - This file
```



Important Notes

Database

- **Do NOT use SQLite in production** - Only PostgreSQL
- Run migrations after deployment: `npx prisma migrate deploy`
- Vercel Postgres is easiest for beginners
- External providers (Supabase, Railway, Neon) are good alternatives

Environment Variables

- **Never commit** `.env` files - They're gitignored

- Use `.env.example` as reference
- Set all variables in Vercel dashboard
- Different secrets for production vs preview

Security

- Generate strong `NEXTAUTH_SECRET` (32+ characters)
- Use environment variables for all secrets
- Keep GitHub Personal Access Token secure
- Rotate secrets periodically

Build

- First build may fail (missing env variables - expected)
- Check build logs in Vercel dashboard
- Prisma generation happens automatically

Success Criteria

Your deployment is successful when:

- ✓ Code is pushed to GitHub
- ✓ Vercel build completes without errors
- ✓ Application is accessible at your Vercel URL
- ✓ Database connection works
- ✓ Users can authenticate (login/register)
- ✓ All features work as expected
- ✓ No console errors
- ✓ HTTPS is working (automatic with Vercel)

Support Resources

Documentation Files (In This Project)

- `VERCEL_DEPLOYMENT_GUIDE.md` - Main deployment guide
- `VERCEL_ENV_VARIABLES.md` - Environment variables reference
- `TROUBLESHOOTING.md` - Common issues and solutions
- `GITHUB_PUSH_INSTRUCTIONS.md` - GitHub authentication help

External Resources

- [Vercel Documentation](https://vercel.com/docs) (https://vercel.com/docs)
 - [Next.js Deployment](https://nextjs.org/docs/deployment) (https://nextjs.org/docs/deployment)
 - [Prisma with Vercel](https://www.prisma.io/docs/guides/deployment/deployment-guides/deploying-to-vercel) (https://www.prisma.io/docs/guides/deployment/deployment-guides/deploying-to-vercel)
 - [NextAuth.js Docs](https://next-auth.js.org/configuration/options) (https://next-auth.js.org/configuration/options)
 - [GitHub Authentication](https://docs.github.com/en/authentication) (https://docs.github.com/en/authentication)
-

You're Almost There!

Your app is fully configured and ready for deployment! Just follow these final steps:

1. **Push to GitHub** (see `GITHUB_PUSH_INSTRUCTIONS.md`)
2. **Import to Vercel** (see `VERCEL_DEPLOYMENT_GUIDE.md`)
3. **Configure environment variables**
4. **Deploy and test**

Good luck with your deployment! 🚀

Quick Command Reference

```
# Push to GitHub
cd /home/ubuntu/studio_2_app/nextjs_space
git push -u origin master

# After Vercel setup
vercel login
vercel link
vercel env pull .env.local
npx prisma migrate deploy

# Test locally with Vercel environment
vercel dev

# View logs
vercel logs --follow
```

Deployment Prepared: October 21, 2025

Application: Studio 2 - Research, Coaching & Counseling Platform

Repository: <https://github.com/Stefanbotes/Studiov2.git>

Target Platform: Vercel