

Render Build Command Update

Current Issue

Build is failing silently after npm install with no error output from prisma generate or npm run build.

Solution

Update the Render build command to use our verbose build script.

Steps to Update Build Command

Option 1: Use render.yaml (Recommended)

The `render.yaml` file is now in the repository. Render should automatically detect and use it.

If Render doesn't auto-detect:

1. Go to Render dashboard
2. Select “carelinkai” service
3. Go to Settings
4. Look for “Use render.yaml” or similar option
5. Enable it

Option 2: Manual Build Command Update

1. Go to Render dashboard: <https://dashboard.render.com>

2. Select “carelinkai” service

3. Click “Settings” in left sidebar

4. Find “Build Command” field

5. Change from:

```
npm install && npx prisma generate && npm run build
```

To:

```
bash render-build-verbose.sh
```

6. Click “Save Changes”

7. Go back to service overview

8. Click “Manual Deploy” → “Deploy latest commit”

Option 3: Inline Verbose Command

If you prefer not to use a script file, use this inline command:

```
set -ex && echo "Installing..." && npm install --legacy-peer-deps && echo "Prisma..." && npx prisma generate && echo "Building..." && npm run build && echo "Done!"
```

What This Will Show

With the verbose script, you'll see:

- Clear step indicators

- Exit codes for each step
- Exact point of failure
- Better error messages
- Timing information

Expected Output

```
=====
STEP 1: INSTALL DEPENDENCIES
=====

+ npm install --legacy-peer-deps
added 1555 packages in 51s
 npm install completed successfully

=====
STEP 2: GENERATE PRISMA CLIENT
=====

+ npx prisma generate
Prisma schema loaded from prisma/schema.prisma
 prisma generate completed successfully

=====
STEP 3: BUILD NEXT.JS APPLICATION
=====

+ npm run build
> carelinkai@0.1.0 build
> next build
 npm run build completed successfully

=====
BUILD COMPLETED SUCCESSFULLY!
=====
```

If Build Still Fails

The verbose output will show exactly which step fails and why.

Common issues:

1. **Prisma generate fails** → Database connection or schema issue
2. **npm run build fails** → TypeScript errors or build configuration issue
3. **Memory limit** → Need to upgrade Render plan

Troubleshooting

If you see “prisma: not found”

- Prisma is not installed correctly
- Try: `npm install prisma @prisma/client --save-dev`

If you see “out of memory”

- Render free tier has memory limits
- Upgrade to paid plan or optimize build

If you see TypeScript errors

- Fix the errors in code
 - Or add `"skipLibCheck": true` to tsconfig.json
-

Status: Ready to deploy with verbose logging

Action: Update build command in Render dashboard