

# Monitoring Render Deployment

---

## Current Status

-  **GitHub Push:** Complete (commit 1f59d9b )
  -  **Render Detection:** Waiting (auto-detect within 2-5 minutes)
  -  **Build Process:** Not started yet
  -  **Deployment:** Pending
- 

## How to Monitor

### Option 1: Render Dashboard (Recommended)

1. Go to: <https://dashboard.render.com/web/srv-d3iso13uibrs73d5fm1g>
2. Look for a new deployment with commit 1f59d9b
3. Click on the deployment to view live logs

### Option 2: GitHub Webhooks

1. Go to: <https://github.com/profyt7/carelinkai/settings/hooks>
2. Check if Render webhook is configured
3. Look for recent delivery status

### Option 3: Check Deployment History

1. Go to: <https://dashboard.render.com/web/srv-d3iso13uibrs73d5fm1g/deploy>
  2. Look for the latest deployment
  3. Expected commit: 1f59d9b
  4. Expected commit message: "Add database migrations to render build script"
- 

## What to Look For in Logs

### 1. Build Start

```
==> Cloning from https://github.com/profyt7/carelinkai
==> Checking out commit 1f59d9b in branch main
```

### 2. Migration Execution (NEW!)

```
echo "STEP 2.5: RUN DATABASE MIGRATIONS"
node_modules/.bin/prisma migrate deploy
```

### 3. Expected Migration Output

```

Prisma schema loaded from prisma/schema.prisma
Datasource "db": PostgreSQL database

2 migrations found in prisma/migrations

Applying migration 20251208_create_placement_search
Applying migration 20251208_create_placement_request

The following migration(s) have been applied:
migrations/
  20251208_create_placement_search/
    migration.sql
  20251208_create_placement_request/
    migration.sql

 All migrations have been successfully applied.

```

### 4. Success Indicators

- “npm install completed successfully”
- “prisma generate completed successfully”
- “All migrations have been successfully applied”
- “Build complete”
- “Deploy complete”

### 5. Error Indicators to Watch For

- “Migration failed”
  - “relation ‘PlacementSearch’ already exists”
  - “Database connection error”
  - “Build failed”
-

## Timeline Expectations

Phase	Expected Duration	Status
GitHub Webhook	1-2 minutes	⌚ Waiting
Render Detection	2-5 minutes	⌚ Waiting
Build Start	Immediate	⌚ Pending
npm install	30-60 seconds	⌚ Pending
Prisma generate	15-30 seconds	⌚ Pending
<b>Database Migrations</b>	<b>10-30 seconds</b>	⌚ Pending
Next.js Build	3-5 minutes	⌚ Pending
Deployment	30-60 seconds	⌚ Pending
<b>TOTAL</b>	<b>~7-12 minutes</b>	-

## Verification Steps (After Deployment)

### 1. Check Database Tables

```
-- Connect to production database
-- Run these queries:

SELECT COUNT(*) FROM "PlacementSearch";
SELECT COUNT(*) FROM "PlacementRequest";

-- Should return 0 (empty tables, but tables exist)
```

### 2. Test API Endpoints

```
# Test discharge planner search endpoint
curl -X POST https://carelinkai.onrender.com/api/discharge-planner/search \
-H "Content-Type: application/json" \
-d '{"query": "Need assisted living in Brooklyn for 85yo with dementia"}'
```

### 3. Check Frontend Access

1. Navigate to: <https://carelinkai.onrender.com/discharge-planner>
2. Try to access the search form
3. Submit a test search query

## 4. Verify Logs

```
# In Render dashboard, check for:  
- No migration errors  
- No 500 errors  
- Successful API calls
```

---

## Troubleshooting

### If Deployment Doesn't Start After 10 Minutes

1. Check GitHub webhook status
2. Manually trigger deployment from Render dashboard
3. Check Render service settings for auto-deploy configuration

### If Migration Fails

1. Check if tables already exist
2. Review migration SQL files
3. Consider manual migration rollback
4. Re-run deployment

### If Build Fails

1. Review full build logs
  2. Check for dependency issues
  3. Verify environment variables
  4. Check Node.js version compatibility
-

## Key Changes in This Deployment

### render-build.sh

```
# ADDED: Step 2.5 - Run Database Migrations
echo "STEP 2.5: RUN DATABASE MIGRATIONS"
echo "====="
echo "Running Prisma migrations..."
node_modules/.bin/prisma migrate deploy || npx prisma migrate deploy
EXIT_CODE=$?

if [ $EXIT_CODE -ne 0 ]; then
  echo "⚠️ prisma migrate deploy failed, trying prisma db push..."
  node_modules/.bin/prisma db push --accept-data-loss || npx prisma db push --accept-
data-loss
  EXIT_CODE=$?

  if [ $EXIT_CODE -ne 0 ]; then
    echo "❌ Database sync failed with exit code: $EXIT_CODE"
    exit 1
  fi
fi

echo "✅ Database migrations completed successfully"
```

### What This Does

1. Runs `prisma migrate deploy` to apply pending migrations
2. If that fails, falls back to `prisma db push`
3. Creates `PlacementSearch` and `PlacementRequest` tables
4. Ensures database schema matches Prisma models

## Contact Information

**Deployment Date:** December 30, 2025

**Commit ID:** 1f59d9b

**Feature:** AI Discharge Planner Database Schema

**Priority:** High

**Last Updated:** December 30, 2025 14:40 UTC

**Next Check:** In 5 minutes (14:45 UTC)