

Performance Plan Deployment - December 11, 2025

✓ Deployment Trigger Complete

What Was Done

1. **Plan Upgrade:** You upgraded Render to the **Performance Plan** (64GB RAM)
2. **Deployment Trigger:** Added deployment marker to README.md
3. **GitHub Push:** Successfully pushed commit `d07f41d` to `profyt7/carelinkai` (main branch)
4. **Auto-Deploy:** Render will automatically detect the push and redeploy with new resources

Commit Details

- **Commit SHA:** `d07f41d`
- **Message:** chore: Trigger redeploy with Performance plan (64GB RAM)
- **Changed Files:** README.md (harmless deployment marker added)
- **Previous Commit:** `cc54bbf`

Next Steps

1. Monitor Render Deployment

1. Go to your [Render Dashboard](https://dashboard.render.com) (<https://dashboard.render.com>)
2. Navigate to your `carelinkai` service
3. Watch the deployment progress in real-time
4. Look for the new deploy starting with commit `d07f41d`

2. Expected Deployment Timeline

- **Build Phase:** 5-10 minutes (installing dependencies, building Next.js app)
- **Deploy Phase:** 2-3 minutes (starting service with 64GB RAM)
- **Health Checks:** 1-2 minutes (verifying app is responding)
- **Total:** ~10-15 minutes

3. Verify Performance Upgrade

Once deployment completes, check:

- ✓ Service is running with 64GB RAM allocation
- ✓ No memory-related errors in logs
- ✓ Application loads successfully at <https://carelinkai.onrender.com>
- ✓ Database migrations completed successfully
- ✓ All features functioning (especially Documents tab with “Failed to load documents” error)

4. Monitor for Issues

Watch Render logs for:

- ✓ Database connection success
- ✓ Prisma client generation
- ✓ Next.js build completion
- ✓ Health check passes

- X Any migration errors (should be resolved with 64GB RAM)
- X Memory allocation warnings

Rollback Plan

If deployment fails:

1. Check Render logs for specific errors
2. Verify database connection string is valid
3. Ensure all environment variables are set correctly
4. If needed, manually trigger redeploy from Render dashboard
5. Contact Render support if persistent issues

Performance Benefits

With 64GB RAM, you should see:

- ✓ Faster build times
- ✓ No memory-related deployment failures
- ✓ Improved application performance under load
- ✓ Better handling of complex database operations
- ✓ Smoother concurrent request handling

Monitoring Commands

```
# Check Render deployment status (via dashboard)
# - Go to https://dashboard.render.com
# - Select your carelinkai service
# - View "Events" and "Logs" tabs

# Check GitHub push status
cd /home/ubuntu/carelinkai-project
git log --oneline -5
git status

# Verify remote configuration
git remote -v
```

Documentation References

- Project Root: `/home/ubuntu/carelinkai-project`
- GitHub Repo: `profyt7/carelinkai`
- Render URL: `https://carelinkai.onrender.com`
- Render Dashboard: `https://dashboard.render.com`

Previous Deployment Issues

The “Failed to load documents” error in the Documents tab was likely caused by:

1. Memory constraints on the previous plan
2. Database query timeouts
3. Prisma client initialization failures

With 64GB RAM, these issues should be resolved.

Status: PUSH COMPLETE - AWAITING RENDER AUTO-DEPLOY

Last Updated: December 11, 2025

Commit: d07f41d

Status: Pushed to GitHub, Render auto-deploy triggered