

# Inquiries Module Deployment Summary

---


**Date:** December 11, 2025  
**Project:** CareLinkAI  
**GitHub Repo:** profyt7/carelinkai  
**Production URL:** <https://carelinkai.onrender.com>

---

## Deployment Status

---

### Successfully Pushed to GitHub

- **Commit Hash:** 3448113
- **Branch:** main
- **Push Time:** December 11, 2025
- **Status:**  **SUCCESSFUL**

### Trigger Information

- **Method:** GitHub push to trigger Render auto-deploy
  - **Deployment Trigger File:** DEPLOY\_TRIGGER.md updated with timestamp
  - **Expected Action:** Render will automatically detect the push and start deployment
- 

## Changes Included in Deployment

---

### Part 1: UI/UX Improvements (Commit: cc202ca)

- Enhanced inquiry cards with visual status indicators
- Advanced filtering system (status, priority, source)
- Improved mobile responsiveness
- Real-time updates and loading states

### Part 2: CSV Export & Document Upload (Commit: 6c74657)

- CSV export functionality for inquiries data
- Document upload system for lead tracking
- File management interface
- API endpoints for document handling

### Part 3: Analytics Dashboard (Commit: 996c5ff)

- Comprehensive analytics dashboard
- Quick actions for common tasks
- Statistics and metrics tracking
- Visual charts and graphs

### Documentation Added







- INQUIRIES\_MODULE\_PART3\_SUMMARY.md - Complete technical documentation

- `INQUIRIES_MODULE_PART3_SUMMARY.pdf` - PDF version for reference

---

## Render Deployment Process

Render will automatically:

1.  Detect the GitHub push
2.  Pull the latest code from `main` branch
3.  Build the Next.js application
4.  Run database migrations (if any)
5.  Deploy to production environment
6.  Health checks and verification

**Expected Deployment Time:** 5-10 minutes

---

## Monitoring Deployment

### Check Deployment Status

1. Visit Render Dashboard: <https://dashboard.render.com>
2. Navigate to **carelinkai** service
3. Monitor the **Events** tab for deployment progress
4. Check build logs for any errors or warnings

### Verify Deployment Success

After deployment completes (typically 5-10 minutes), verify:

#### 1. **Inquiries Module Access:**

- Navigate to `/operator/inquiries`
- Check if all 3 parts are visible and functional

#### 2. **Key Features to Test:**

- [ ] Inquiry cards display correctly
- [ ] Advanced filters work (status, priority, source)
- [ ] CSV export functionality
- [ ] Document upload/download
- [ ] Analytics dashboard loads
- [ ] Quick actions are functional

#### 3. **API Endpoints:**

- [ ] `/api/inquiries` - List inquiries
  - [ ] `/api/inquiries/[id]` - Get single inquiry
  - [ ] `/api/inquiries/[id]/documents` - Document management
  - [ ] `/api/inquiries/export` - CSV export
  - [ ] `/api/inquiries/analytics` - Analytics data
-



## Known Issues to Monitor

Based on previous error logs, monitor these areas:

1. **Caregiver API (500 Error):**
  - URL: `/api/operator/caregivers`
  - Watch for: Server errors and 405 Method Not Allowed
2. **Document Loading:**
  - Check if “Failed to load documents” persists
  - Verify API responses are returning data correctly
3. **TypeScript Errors:**
  - Monitor for destructuring property errors
  - Check browser console for runtime errors



## Git History

```
3448113 - chore: Deploy Inquiries Module (all 3 parts) to production
996c5ff - feat: Add analytics dashboard and quick actions to inquiries module - Part 3
6c74657 - feat: Add CSV export and document upload to Inquiries Module - Part 2
cc202ca - feat: Add comprehensive UI/UX improvements and advanced filters to
Inquiries Module - Part 1
```



## Rollback Instructions

If issues arise after deployment:

```
# 1. Revert to previous stable commit
git revert 3448113

# 2. Or reset to previous version
git reset --hard 996c5ff

# 3. Force push to trigger rollback
git push origin main --force

# 4. Monitor Render deployment for rollback completion
```



## Next Steps

1. **Wait 5-10 minutes** for Render deployment to complete
2. **Check Render Dashboard** for deployment status
3. **Test Inquiries Module** on production URL
4. **Monitor error logs** in browser console and Render logs
5. **Document any issues** that arise post-deployment

---

## Support & Troubleshooting

---

### If Deployment Fails:

1. Check Render build logs for errors
2. Verify all environment variables are set
3. Check database migrations completed successfully
4. Review API endpoint errors in logs

### If Features Don't Work:







1. Clear browser cache and hard reload
2. Check browser console for JavaScript errors
3. Verify API responses in Network tab
4. Test with different user roles (ADMIN, OPERATOR)

---

## Success Criteria

---

Deployment is considered successful when:

-  Render build completes without errors
-  Application loads at <https://carelinkai.onrender.com>
-  Inquiries Module accessible at `/operator/inquiries`
-  All 3 parts (filters, CSV, analytics) are functional
-  No new console errors appear
-  Existing functionality remains intact

---

**Deployment Initiated By:** DeepAgent

**Commit Reference:** 3448113

**GitHub Push:** Successful 

**Render Auto-Deploy:** Triggered 

**Status:** Awaiting deployment completion...