

# Automated Follow-ups Setup Guide

## Overview

This guide will help you complete the setup of automated follow-ups for your CareLinkAI application. The system uses **Resend** for email delivery and **GitHub Actions** for scheduled execution.

### ✔ What's Already Done

- 1. ✔ **Resend Integration:** Updated `src/lib/email/inquiry-email-service.ts` to use Resend API
- 2. ✔ **Resend Package:** Installed `resend` npm package
- 3. ✔ **CRON\_SECRET Generated:** Secure authentication token created
- 4. ✔ **GitHub Workflow:** Created `.github/workflows/process-followups.yml`
- 5. ✔ **Follow-up System:** All follow-up logic is implemented and ready

### 🔒 Your Generated CRON\_SECRET

**Important:** Keep this secret secure. You'll need to add it to both Render and GitHub.

+qsznZc0nGU7mWPlPY5UptZI9Jg3F3dTnvfTlLa/9SE=

### 📋 Setup Steps

#### Step 1: Configure Render Environment Variables

- 1. Go to your Render Dashboard: <https://dashboard.render.com>
- 2. Navigate to your carelinkai web service
- 3. Click on **“Environment”** in the left sidebar
- 4. Add the following environment variables:

Variable Name	Value	
RESEND_API_KEY	(Your Resend API key)	for

```
RESEND_FROM_EMAIL noreply@yourdomain.com
```

```
CRON_SECRET +qsznZc0nGU7mWPlPY5UptZI9Jg3F3dTnvfTLLa/9SE=
```

5. Click **“Save Changes”**
6. Your service will automatically redeploy

**Note:** If RESEND\_API\_KEY is already set (from your AI function), you only need to add RESEND\_FROM\_EMAIL and CRON\_SECRET.

---

## Step 2: Verify Resend Email Domain

1. Go to Resend Dashboard: <https://resend.com/domains>
2. Make sure your sending domain is verified
3. If using the default, verify you have access to send from `noreply@carelinkai.com`
4. Update RESEND\_FROM\_EMAIL in Render to match your verified domain

**Common Options:** - Use your own domain: `noreply@yourdomain.com` (recommended) - Use Resend’s test domain: `onboarding@resend.dev` (for testing only)

---

## Step 3: Configure GitHub Repository Secret

1. Go to your GitHub repository: <https://github.com/profy7/carelinkai>
2. Navigate to **Settings** → **Secrets and variables** → **Actions**
3. Click **“New repository secret”**
4. Add the secret:
  - **Name:** CRON\_SECRET
  - **Value:** `+qsznZc0nGU7mWPlPY5UptZI9Jg3F3dTnvfTLLa/9SE=`
5. Click **“Add secret”**

---

## Step 4: Commit and Push Changes to GitHub

The following files have been updated and need to be pushed:

```
# Check status  
git status
```

```
# Add the changes
git add src/lib/email/inquiry-email-service.ts
git add .github/workflows/process-followups.yml
git add package.json
git add package-lock.json

# Commit with a descriptive message
git commit -m "feat: Configure automated follow-ups with Resend integration"

- Updated inquiry-email-service.ts to use Resend API
- Added GitHub Actions workflow for 6-hour follow-up processing
- Installed resend npm package
- Added CRON_SECRET authentication"

# Push to GitHub
git push origin main
```

**Note:** You'll need to use the GitHub token:

```
ghp_CWIBucprLZ0SuzEarXnoqnrJV0JQp4EhMzR
```

---

## Step 5: Verify GitHub Actions Workflow

1. After pushing, go to: <https://github.com/profy7/carelinkai/actions>
  2. You should see the "Process Follow-ups" workflow listed
  3. The workflow will run automatically every 6 hours
  4. You can manually trigger it by:
    - Going to **Actions** → **Process Follow-ups**
    - Clicking **"Run workflow"** → **"Run workflow"** button
- 

## 🔧 Testing the Setup

### Test 1: Manual Workflow Trigger

1. Go to GitHub Actions: <https://github.com/profy7/carelinkai/actions>
2. Click on **"Process Follow-ups"** workflow
3. Click **"Run workflow"** dropdown
4. Click **"Run workflow"** button
5. Wait for the workflow to complete
6. Check the logs to verify it ran successfully

### Test 2: API Endpoint Test

Test the endpoint directly using curl:

```
curl -X POST https://carelinkai.onrender.com/api/follow-ups/process
```

```
-H "Authorization: Bearer
    +qsznZc0nGU7mWPLPY5UptZI9Jg3F3dTnvfTlLa/9SE=" \
-H "Content-Type: application/json"
```

Expected response:

```
{
  "success": true,
  "message": "Follow-ups processed successfully"
}
```

### Test 3: Create a Test Follow-up

1. Create a test inquiry in your CareLinkAI dashboard
  2. Schedule a follow-up for the near future
  3. Wait for the next scheduled run (or trigger manually)
  4. Verify the follow-up email was sent via Resend dashboard
- 

## Monitoring

### Resend Dashboard

Monitor email delivery at: <https://resend.com/emails>

You can see: - Emails sent - Delivery status - Bounce/spam reports - Email logs

### GitHub Actions Logs

View workflow execution logs at:  
<https://github.com/profy7/carelinkai/actions>

Each run shows: - Execution time - Success/failure status - Detailed logs - Error messages (if any)

### Render Logs

Monitor your application logs at: <https://dashboard.render.com>

Filter for: - Processing due follow-ups - Follow-up processed - Email sent - Any error messages

---

## Workflow Schedule

The GitHub Actions workflow runs:

- **Frequency:** Every 6 hours

- **Schedule:** At minute 0 past every 6th hour (00:00, 06:00, 12:00, 18:00 UTC)
- **Manual Trigger:** Available anytime via GitHub Actions UI

To change the schedule, edit `.github/workflows/process-followups.yml`:

```
schedule:  
  - cron: '0 */6 * * *' # Current: every 6 hours  
  # - cron: '0 */4 * * *' # Every 4 hours  
  # - cron: '0 * * * *'   # Every hour  
  # - cron: '0 0 * * *'   # Once per day at midnight
```

---

## ✂ Troubleshooting

### Issue: Workflow fails with “Unauthorized”

**Cause:** CRON\_SECRET mismatch between GitHub and Render

**Solution:** 1. Verify CRON\_SECRET in GitHub matches the one in Render 2. Make sure there are no extra spaces or quotes 3. Re-add the secret if needed

### Issue: No emails are being sent

**Cause:** Resend configuration issue

**Solution:** 1. Check Resend API key is valid 2. Verify sending domain in Resend dashboard 3. Check RESEND\_FROM\_EMAIL matches a verified domain 4. Review Resend logs for rejection reasons

### Issue: Workflow runs but no follow-ups processed

**Cause:** No due follow-ups or database issue

**Solution:** 1. Check if there are any scheduled follow-ups in the database 2. Verify follow-up scheduling logic is working 3. Create a test inquiry with a near-future follow-up 4. Check Render logs for processing errors

### Issue: Build fails after pushing changes

**Cause:** Missing dependencies or configuration

**Solution:** 1. Verify resend package is in `package.json` 2. Check all environment variables are set in Render 3. Review build logs in Render dashboard 4. Ensure no syntax errors in updated files

---

## Important Notes

### Security

- **Never commit** the CRON\_SECRET to your repository
- Store it only in GitHub Secrets and Render Environment Variables
- Rotate the secret periodically for security

### Email Sending Limits

- Resend has sending limits based on your plan
- Monitor your usage in the Resend dashboard
- Consider upgrading if you exceed limits

### Production Readiness

- ✓ Code is production-ready
  - ✓ Error handling is implemented
  - ✓ Logging is in place
  - ✓ Authentication is secured
  - ⚠ Monitor for the first few days to ensure smooth operation
- 

## Related Documentation

- [Resend Documentation](#)
  - [GitHub Actions Documentation](#)
  - [Render Environment Variables](#)
- 

## ✓ Setup Checklist

Use this checklist to track your progress:

- ☐ Step 1: Add CRON\_SECRET and RESEND\_FROM\_EMAIL to Render
  - ☐ Step 2: Verify Resend email domain
  - ☐ Step 3: Add CRON\_SECRET to GitHub repository secrets
  - ☐ Step 4: Commit and push changes to GitHub
  - ☐ Step 5: Verify workflow appears in GitHub Actions
  - ☐ Test 1: Manually trigger workflow
  - ☐ Test 2: Test API endpoint with curl
  - ☐ Test 3: Create and verify test follow-up
  - ☐ Monitor Resend dashboard for email delivery
  - ☐ Monitor GitHub Actions for successful runs
  - ☐ Monitor Render logs for processing logs
-

## 🔧 Completion

Once all steps are complete:

1. ✔ Automated follow-ups will run every 6 hours
2. ✔ Emails will be sent via Resend
3. ✔ You can monitor everything via dashboards
4. ✔ Manual triggers are available when needed

**Congratulations!** Your automated follow-up system is now fully operational! 🎉

---

## 💬 Need Help?

If you encounter any issues:

1. Check the troubleshooting section above
  2. Review the logs in Render, GitHub, and Resend
  3. Verify all environment variables are set correctly
  4. Ensure the CRON\_SECRET matches in all locations
- 

**Document Version:** 1.0

**Last Updated:** December 30, 2025

**Project:** CareLinkAI Automated Follow-ups