Trelowen AI - Complete Production Deployment Guide

DEPLOYMENT READY - All Files Configured

Your Trelowen AI system is now fully prepared for production deployment with permanent hosting.

RECOMMENDED HOSTING: Render.com

Why Render?

- **Free Tier**: 512MB RAM, automatic SSL, 750 hours/month
- **Automatic SSL**: HTTPS certificates via Let's Encrypt
- **Easy Deployment**: GitHub integration, zero-config
- **Reliable**: 99.9% uptime, global CDN
- **Node.js Optimized**: Perfect for your TypeScript/Express app

---

STEP-BY-STEP DEPLOYMENT

Step 1: Upload to GitHub

```
# Create a new GitHub repository and upload your code
# The /home/ubuntu/trelowen-ai folder contains all necessary files
```

Step 2: Deploy to Render

1. **Sign up**: Go to https://render.com
2. **Connect GitHub**: Link your GitHub account
3. **Create Web Service**:
   - Click "New +" → "Web Service"
   - Connect your GitHub repository
   - **Build Command**: `npm install && npm run build`
   - **Start Command**: `npm start`
   - **Plan**: Free

Step 3: Configure Environment Variables

In Render Dashboard → Environment, add these variables:

```
NODE_ENV=production
PORT=10000
OPENAI_API_KEY=sk-admin-Q87LScrPWs13ndgnaSvWBv4PZIlVD6EBfRd-
D0_ivxVIL704bgaoXGot3EyT3BlbkFJXDwQcIqiR1eP-iHkYgc1qeRIRurecdWbpGV1s2mtFHgBu_ZDweq9vtV-
kA
OPENAI_MODEL=gpt-4-turbo-preview
OPENAI_EMBEDDING_MODEL=text-embedding-3-small
JWT_SECRET=trelowen_jwt_secret_production_2024
WEBHOOK_SECRET=trelowen_webhook_secret_production_2024

# Optional: Add these when you have the credentials
GHL_API_KEY=your_ghl_api_key_here
GHL_LOCATION_ID=your_ghl_location_id_here
TWILIO_ACCOUNT_SID=your_twilio_account_sid
TWILIO_AUTH_TOKEN=your_twilio_auth_token
TWILIO_PHONE_NUMBER=your_twilio_phone_number
SMTP_HOST=smtp.gmail.com
SMTP_PORT=587
SMTP_USER=your_email@gmail.com
SMTP_PASS=your_app_password
```

YOUR PRODUCTION URLS

After deployment, you'll get:
- **API Base**: `https://trelowen-ai-[random].onrender.com`
- **Health Check**: `https://trelowen-ai-[random].onrender.com/api/health`
- **Chat API**: `https://trelowen-ai-[random].onrender.com/api/chat`
- **Widget**: `https://trelowen-ai-[random].onrender.com/widget/widget.js`

UPDATE LOVEABLE WEBSITE

Replace Current Script

Remove the current localhost script and add this production version:

```
<!-- Trelowen AI Chat Widget - Production -->
<script>
  (function() {
    var script = document.createElement('script');
    script.src = 'https://trelowen-ai-[YOUR-URL].onrender.com/widget/widget.js';
    script.async = true;
    script.onload = function() {
      console.log('Trelowen AI Widget loaded successfully');
    };
    document.head.appendChild(script);
  })();
</script>
```

⚠ **Important**: Replace `[YOUR-URL]` with your actual Render URL after deployment.

TESTING YOUR DEPLOYMENT

1. Health Check

```
curl https://your-app-url.onrender.com/api/health
```

Expected Response: `{"status":"healthy","timestamp":"...","service":"Trelowen AI Assistant"}`

2. Chat API Test

```
curl -X POST https://your-app-url.onrender.com/api/chat \
  -H "Content-Type: application/json" \
  -d '{"message":"Hello, tell me about Trelowen"}'
```

3. Widget Test

- Open your Loveable website
- Check browser console for “Trelowen AI Widget loaded successfully”
- Click the chat button to test functionality

SECURITY FEATURES CONFIGURED

- **HTTPS/SSL**: Automatic certificates
- **CORS Protection**: Configured for Loveable domains
- **Rate Limiting**: 100 requests per 15 minutes per IP
- **Security Headers**: Helmet.js protection
- **Environment Variables**: Secure credential storage

CUSTOM DOMAIN (OPTIONAL)

Add Your Own Domain

1. **In Render Dashboard**: Settings → Custom Domains
2. **Add Domain**: e.g., `api.trelowen.com`
3. **Update DNS**: Add CNAME record as instructed
4. **SSL**: Automatically provisioned

Update Widget Script

```
<script>
  (function() {
    var script = document.createElement('script');
    script.src = 'https://api.trelowen.com/widget/widget.js';
    script.async = true;
    document.head.appendChild(script);
  })();
</script>
```

MONITORING & MAINTENANCE

Render Dashboard Features

- **Logs**: Real-time application logs
- **Metrics**: CPU, memory, response times
- **Deployments**: Automatic from GitHub pushes
- **Scaling**: Upgrade plans as needed

Health Monitoring

- **Endpoint**: `/api/health`
- **Auto-restart**: On failures
- **Uptime**: 99.9% SLA

TROUBLESHOOTING

Common Issues & Solutions

1. Widget Not Loading
- Check CORS configuration in production
- Verify script URL is correct
- Check browser console for errors

2. API Errors
- Verify environment variables are set
- Check Render deployment logs
- Test health endpoint first

3. OpenAI API Issues
- Verify API key is valid
- Check usage limits
- Monitor rate limiting

COST BREAKDOWN

Render Free Tier

- **Cost**: $0/month
- **Includes**: 512MB RAM, 750 hours, SSL, custom domains
- **Limitations**: Sleeps after 15 minutes of inactivity

Upgrade Options

- **Starter**: $7/month (always-on, more resources)
- **Standard**: $25/month (autoscaling, priority support)

SUPPORT & NEXT STEPS

Immediate Actions

1. Deploy to Render using the instructions above
2. Update Loveable website with production script
3. Test all functionality
4. Monitor for 24-48 hours

Future Enhancements

• Add custom domain for branding
• Implement analytics tracking
• Set up monitoring alerts
• Scale to paid plan if needed

DEPLOYMENT CHECKLIST

• [ ] GitHub repository created and code uploaded
• [ ] Render account created and service deployed
• [ ] Environment variables configured
• [ ] Health check endpoint tested
• [ ] Chat API functionality verified
• [ ] Loveable website script updated
• [ ] Widget functionality tested on live site
• [ ] CORS working for cross-origin requests
• [ ] SSL certificate active and working
• [ ] Monitoring and logs reviewed

Your Trelowen AI system is ready for permanent, reliable hosting!

Estimated Setup Time: 15-30 minutes
Total Cost: Free (Render free tier)
Uptime: 99.9% guaranteed
SSL: Automatic HTTPS
Support: 24/7 platform monitoring

For technical support or questions about this deployment, refer to the detailed logs and documentation provided.