

Workigom Render Deployment - Ready for Push

Deployment Configuration Complete!

All necessary files have been created and configured for Render.com deployment. Your project is now ready to be deployed!

What Was Created/Updated

New Files

1. `render.yaml` - Render Blueprint configuration for one-click deployment
2. `RENDER_DEPLOYMENT.md` - Comprehensive deployment guide (13KB)
3. `RENDER_QUICK_REFERENCE.md` - Quick reference guide
4. `render-build.sh` - Build automation script (executable)
5. `.renderignore` - Files to exclude from deployment

Updated Files

1. `README.md` - Added Render deployment section
 2. `backend/.env.example` - Enhanced with comprehensive documentation
 3. `src-frontend/.env.example` - Enhanced with deployment examples
-

Action Required: Push to GitHub

Due to GitHub authentication limitations in the current session, you need to manually push the changes to GitHub.

Step 1: Push Changes

Open a terminal and run:

```
cd /path/to/workigom

# Check status
git status

# If changes are already committed (they should be):
git push origin master:main
```

Or if you need to commit first:

```
git add .
git commit -m "feat: Add Render deployment configuration"
git push origin master:main
```

Step 2: Verify on GitHub

1. Go to: <https://github.com/volkanakbulut73/Workigom>

2. Verify that the following files are present:

- `render.yaml`
 - `RENDER_DEPLOYMENT.md`
 - `RENDER_QUICK_REFERENCE.md`
 - `render-build.sh`
 - `.renderignore`
-



Next Steps: Deploy on Render

Once the files are pushed to GitHub, follow these steps:

Quick Deploy (5 minutes)

1. Go to Render Dashboard

- Visit: <https://dashboard.render.com/>
- Sign in or create an account

2. Deploy with Blueprint

...

3. Click “New +” → “Blueprint”

4. Select: volkanakbulut73/Workigom

5. Render will detect `render.yaml`

6. Click “Apply”

7. Wait 5-10 minutes for deployment

...

8. Run Database Migrations

`bash`

```
# After backend is deployed, open backend shell and run:  
npx prisma migrate deploy
```

9. Update Environment Variables

- Backend: Update `CORS_ORIGIN` with your frontend URL
- Frontend: Update `VITE_BACKEND_URL` with your backend URL

10. Redeploy Services

- Services will auto-redeploy after env var changes



Configuration Summary

Backend Service (workigom-backend)

- **Type:** Web Service (Node.js)
- **Root Directory:** `backend/`
- **Build Command:** `npm install && npx prisma generate && npm run build`

- **Start Command:** `npm run start`
- **Health Check:** `/api/health`
- **Environment Variables:** Auto-configured via `render.yaml`

Frontend Service (workigom-frontend)

- **Type:** Static Site
- **Root Directory:** `src-frontend/`
- **Build Command:** `npm install && npm run build`
- **Publish Directory:** `dist/`
- **Routing:** SPA routing configured

Database (workigom-db)

- **Type:** PostgreSQL
- **Plan:** Free
- **Region:** Frankfurt
- **Auto-linked:** Backend receives `DATABASE_URL` automatically

Environment Variables Reference

Backend Required Variables

```
NODE_ENV=production
PORT=10000
DATABASE_URL=<auto-configured>
JWT_SECRET=<auto-generated>
JWT_REFRESH_SECRET=<auto-generated>
CORS_ORIGIN=<update-with-frontend-url>
```

Frontend Required Variables

```
VITE_BACKEND_URL=<update-with-backend-url>
```

Documentation Available

1. **RENDERS_DEPLOYMENT.md** (`./RENDERS_DEPLOYMENT.md`)
 - Complete step-by-step guide
 - Troubleshooting section
 - Post-deployment checklist
 - ~13KB of detailed instructions
2. **RENDERS_QUICK_REFERENCE.md** (`./RENDERS_QUICK_REFERENCE.md`)
 - Quick command reference
 - Common issues and solutions
 - Service URLs format

3. **README.md** (./README.md)

- Updated with Render section
 - Links to all guides
 - Build script usage
-

Pre-Deployment Checklist

- [x] render.yaml configured with all services
 - [x] Backend build and start commands set
 - [x] Frontend build commands set
 - [x] Database configuration added
 - [x] Environment variables documented
 - [x] Health check endpoint configured
 - [x] CORS settings prepared
 - [x] .env.example files updated
 - [x] Deployment guides created
 - [x] Build scripts created and tested
 - [] **Changes pushed to GitHub ← YOU ARE HERE**
 - [] Blueprint deployed on Render
 - [] Database migrations run
 - [] Environment variables updated with actual URLs
 - [] Services tested and verified
-

Manual Build Testing (Optional)

Before deploying to Render, you can test the build locally:

```
# Test backend build  
./render-build.sh backend  
  
# Test frontend build  
./render-build.sh frontend  
  
# Test both  
./render-build.sh all
```

Troubleshooting

If Push Fails

```
# Ensure you're authenticated
git config user.name "Your Name"
git config user.email "your@email.com"

# Try force push (if safe)
git push -f origin master:main
```

If Render Deployment Fails

1. Check build logs in Render dashboard
2. Verify environment variables are set
3. Ensure database is provisioned
4. Check RENDER_DEPLOYMENT.md troubleshooting section

Support Resources

- **Render Documentation:** <https://render.com/docs>
- **Render Community:** <https://community.render.com>
- **Project Deployment Guide:** [RENDER_DEPLOYMENT.md](#) (./RENDER_DEPLOYMENT.md)
- **Quick Reference:** [RENDER_QUICK_REFERENCE.md](#) (./RENDER_QUICK_REFERENCE.md)

Git Commit Summary

The following commit is ready to push:

```
feat: Add comprehensive Render.com deployment configuration

- Updated render.yaml with improved service definitions and auto-deploy
- Created detailed RENDER_DEPLOYMENT.md guide with step-by-step instructions
- Added RENDER_QUICK_REFERENCE.md for quick lookups
- Created render-build.sh script for manual builds
- Updated backend and frontend .env.example files
- Added .renderignore file
- Updated README.md with Render deployment instructions
```

All services ready **for** one-click deployment via Render Blueprint!

Files Changed:

- Modified: README.md, render.yaml
- Modified: backend/.env.example, src-frontend/.env.example
- Created: .renderignore, RENDER_DEPLOYMENT.md, RENDER_QUICK_REFERENCE.md, render-build.sh



Summary

Your Workigom project is fully configured for Render deployment!

Next Action: Push the changes to GitHub using the command above, then follow the [RENDER_DEPLOYMENT.md](#) (./RENDER_DEPLOYMENT.md) guide to deploy.

Expected Deployment Time: 5-10 minutes for initial deployment

Services After Deployment:

- Backend API: <https://workigom-backend.onrender.com>
 - Frontend App: <https://workigom-frontend.onrender.com>
 - PostgreSQL Database: (Internal connection string)
-

Prepared: November 1, 2024

Status:  Ready for GitHub Push → Render Deployment

Configuration Version: 1.0.0