



Problem Fixed! - Immediate Next Steps

🎯 What Was the Problem?

Railway was **ignoring your Root Directory setting** (`backend/`) because there was a `railway.json` file in the root directory that was overriding your dashboard settings.

This root config file was telling Railway to:

- ❌ Use the frontend Dockerfile (with nginx)
- ❌ Ignore the backend directory
- ❌ Deploy as a frontend app



What We Fixed

1. Removed the Conflicting File

```
❌ /home/ubuntu/workigom/railway.json (deleted)
✓  /home/ubuntu/workigom/railway.json.frontend.backup (backed up)
```

2. Created Comprehensive Documentation

- `RAILWAY_ROOT_DIRECTORY_FIX.md` - Complete troubleshooting guide
- Explains Railway config precedence
- Provides multiple solution options
- Includes debugging steps

3. Committed & Pushed to GitHub

```
Commit: 7c4bf8d
Message: "Fix: Remove root railway.json to fix backend deployment"
Branch: master
Status: Pushed ✓
```



Next Steps (DO THIS NOW!)

Step 1: Trigger Railway Redeploy

Go to your Railway dashboard and redeploy:

Option A: Quick Redeploy

1. Open <https://railway.app>
2. Navigate to your `workigom` service
3. Click on “Deployments” tab
4. Find the latest deployment

5. Click the “ : ” (three dots) menu

6. Select “**Redeploy**”

Option B: Wait for Auto-Deploy

- Railway should automatically detect the GitHub push
 - Watch for a new deployment to start (within 1-2 minutes)
-

Step 2: Verify Correct Dockerfile

Once deployment starts, check the logs:

✓ Signs of SUCCESS (Backend Dockerfile):

```
Using Detected Dockerfile
FROM docker.io/library/node:20-alpine
COPY package*.json ./
COPY prisma ./prisma/
RUN npm run prisma:generate
Running Prisma migrations...
🚀 Starting Workigom Backend...
✅ Starting server...
```

✗ Signs of FAILURE (Still using Frontend):

```
FROM nginx:alpine
COPY nginx.conf /etc/nginx/conf.d/default.conf
ERROR: "/nginx.conf": not found
```

Step 3: If It Still Fails - Nuclear Option

If Railway is still caching the old config:

1. Disconnect the Service:

- Go to Service Settings
- Scroll to “Danger Zone”
- Click “Disconnect Source Repository”
- Confirm

2. Wait 30 seconds

3. Reconnect the Service:

- Click “Connect Repository”
 - Select `volkanakbulut73/workigom`
 - **IMPORTANT:** Set “Root Directory” to `backend/`
 - Select branch: `master`
 - Click “Deploy”
-

Verification Checklist

After deployment completes:

1. Check Deployment Logs

- [] Logs show Node.js base image (not nginx)
- [] Logs show Prisma commands
- [] Logs show “Starting Workigom Backend”
- [] No nginx.conf errors

2. Test Health Endpoint

```
curl https://your-backend-url.railway.app/api/health
```

Expected response:

```
{
  "status": "ok",
  "timestamp": "2025-10-23T...",
  "uptime": 123.45,
  "database": "connected"
}
```

3. Check Railway Dashboard

- [] Service status: Running (green)
- [] Deployment status: Success
- [] No error alerts

Current Configuration

Now that the root `railway.json` is removed, Railway will use:

Priority 1: Dashboard Settings

- Root Directory: `backend/`
- Builder: Dockerfile
- Branch: master

Priority 2: Backend Config Files

Either of these (Railway checks in order):

- `backend/railway.json` - Exists 
- `backend/railway.toml` - Exists 

Priority 3: Backend Dockerfile

- `backend/Dockerfile` - Exists 
- Uses Node.js (not nginx)
- Includes Prisma migrations
- Has health check

- Runs start.sh script
-

Expected Outcome

Once redeployed with the fix:

1. ✓ Railway reads Dashboard setting: Root Directory = backend/
 2. ✓ Railway changes working directory to backend/
 3. ✓ Railway finds backend/railway.json
 4. ✓ Railway uses backend/Dockerfile
 5. ✓ Build includes Prisma client generation
 6. ✓ Deployment runs database migrations
 7. ✓ Backend starts on port 3001
 8. ✓ Health check passes
 9. ✓ Your backend is LIVE! 
-

Before vs After

BEFORE (Broken):

```
Repository Root
└── railway.json ⚠️ (overrides everything)
    └── dockerfilePath: "Dockerfile" (frontend!)
└── Dockerfile (frontend with nginx)
└── backend/
    ├── railway.json (ignored!)
    ├── railway.toml (ignored!)
    └── Dockerfile (ignored!)
```

Railway used: ./Dockerfile (frontend) ❌

AFTER (Fixed):

```
Repository Root
└── railway.json.frontend.backup (disabled)
└── backend/
    ├── railway.json ✓ (now active!)
    ├── railway.toml ✓ (backup)
    └── Dockerfile ✓ (backend!)
```

Railway uses: backend/Dockerfile (backend) ✓

Troubleshooting

Issue: Railway still using wrong Dockerfile

Solution:

```
# Check what was pushed to GitHub
cd /home/ubuntu/workigom
git log --oneline -5
git show HEAD --name-status

# Verify root railway.json is gone
ls -la railway.json 2>/dev/null || echo "✓ File removed"

# Verify backend configs exist
ls -la backend/railway.json
ls -la backend/Dockerfile
```

If files are correct but Railway still fails:

- Clear browser cache
- Use Railway CLI: `railway up --service workigom`
- Or disconnect/reconnect service (nuclear option above)

Issue: Can't find deployment logs

1. Go to Railway dashboard
2. Click on your `workigom` service (the one showing “Failed”)
3. Click on “Deployments” tab
4. Click on the latest deployment
5. Logs should appear on the right side

Issue: Authentication errors

Make sure environment variables are set:

- `DATABASE_URL` - PostgreSQL connection string
- `JWT_SECRET` - Secret for JWT tokens
- `PORT` - Usually 3001 (optional)

What to Report Back

After you trigger the redeploy, let me know:

1. Deployment Status:

- Did a new deployment start automatically?
- Or did you manually trigger it?

2. First 20 Lines of Logs:

- Does it show Node.js base image?
- Does it show Prisma commands?

3. Final Result:

- Did deployment succeed? (green checkmark)
- Or still failing? (red X)

4. Health Check:

- Can you access `/api/health` ?
- What's the response?



Documentation Created

For future reference:

- `RAILWAY_ROOT_DIRECTORY_FIX.md` - Complete troubleshooting guide (23 KB)
 - `IMMEDIATE_ACTIONS.md` - This file (quick reference)
 - `backend/railway.toml` - Updated with clarifying comments
 - `railway.json.frontend.backup` - Original file (backed up, not deleted)
-

✨ Key Takeaway

Railway Configuration Precedence:

1. `railway.json` in REPO ROOT (highest - was causing issue)
2. `railway.toml` in REPO ROOT
3. Dashboard Settings (Root Directory)
4. `railway.json` in Root Directory folder
5. `railway.toml` in Root Directory folder
6. Auto-detection (lowest)

By removing the root `railway.json`, we let your dashboard settings take control! 🚀

Status: ✅ Fix Applied & Committed

Commit: 7c4bf8d

Branch: master

Action Required: Trigger Railway redeploy (see Step 1 above)

Next: Watch the deployment logs and verify it's using the backend Dockerfile! 🚀