

✓ GitHub Push Successful - Railway Deployment Next Steps

🎉 What Was Completed

1. SSH Authentication ✓

- Successfully authenticated with GitHub using SSH
- Connection test passed: `Hi volkanakbulut73/workigom! You've successfully authenticated`

2. Git Configuration Updated ✓

- Changed remote URL from HTTPS to SSH
- **Old:** `https://ghu_***@github.com/volkanakbulut73/workigom.git`
- **New:** `git@github.com:volkanakbulut73/workigom.git`

3. Railway Configuration Created ✓

- Created `railway.toml` file at project root
- Configuration:

```
toml
[build]
builder = "dockerfile"
dockerfilePath = "backend/Dockerfile"
```

4. Changes Committed and Pushed ✓

- Commit: `ed54bb0 - Add railway.toml to configure backend Dockerfile deployment`
- Pushed to: `master` branch
- Repository: `volkanakbulut73/workigom`

5. Backend Dockerfile Verified ✓

- Marker comment present: “🎯 WORKIGOM BACKEND DOCKERFILE - If you see this in Railway logs, the correct Dockerfile is being used! 🎯”
- Located at: `backend/Dockerfile`

🚀 Next Steps for Railway Deployment



Step 1: Trigger Railway Rebuild

Railway should automatically detect the new commit and trigger a rebuild. If not:

1. Go to your Railway dashboard
2. Navigate to the **workigom** service
3. Click on the **Deployments** tab
4. Click **Deploy** to manually trigger a new deployment

Step 2: Monitor Build Logs

CRITICAL: Watch the build logs for the verification marker:

1. Click on the latest deployment
2. Look at the build logs
3. **You should see:** #  WORKIGOM BACKEND DOCKERFILE - If you see this in Railway logs, the correct Dockerfile is being used! 

✓ **If you see this marker** → Railway is using the correct backend Dockerfile

✗ **If you don't see it** → Railway is still using the wrong Dockerfile

Step 3: Verify Root Directory Setting (If Build Still Fails)

According to your screenshots, you have Root Directory set to “backend”. This might be causing conflicts with the railway.toml configuration.

Try These Options:

Option A: Remove Root Directory (Recommended)

1. Go to workigom service → **Settings** tab
2. Find **Root Directory** setting
3. **CLEAR/REMOVE** the “backend” value (leave it empty)
4. Save and redeploy
5. Railway will use railway.toml to find the Dockerfile

Option B: Keep Root Directory BUT Update railway.toml

If you want to keep “backend” as root directory:

1. Update railway.toml to:

```
toml
[build]
builder = "dockerfile"
dockerfilePath = "Dockerfile" # Changed from "backend/Dockerfile"
```

2. Commit and push
3. Redeploy

I recommend Option A - let railway.toml handle everything.

Step 4: Check for These Common Issues

1. **Wrong Branch Connected**
 - Verify “Branch connected to production” is set to `master`
 - Should show in Settings tab
2. **Environment Variables Missing**
 - Check if all required environment variables are set
 - Backend typically needs: DATABASE_URL, JWT_SECRET, etc.
3. **Port Configuration**
 - Railway expects port from \$PORT environment variable
 - Verify backend is using: `process.env.PORT || 3000`

Step 5: Success Indicators

✓ Build should succeed if you see:

- The marker comment in logs
- Build completing without “nginx.conf not found” error
- Backend dependencies installing (npm ci)
- Prisma client generation succeeding

✓ Deployment should succeed if you see:

- Container starting
- Backend server listening on port
- Health checks passing (if configured)



Troubleshooting

If Build Still Uses Wrong Dockerfile:

1. Double-check railway.toml is at project root

```
bash
```

```
git show master:railway.toml
```

2. Verify Railway detected the change

- Railway dashboard should show new commit
- Check commit hash matches: `ed54bb0`

3. Force Railway to re-read configuration

- Disconnect and reconnect GitHub repository
- Or make a small change and push again

If Build Fails with Different Error:

1. Check the specific error message
2. Review backend/Dockerfile for any missing files
3. Verify all COPY commands reference correct paths
4. Check .dockerignore isn't excluding needed files



Current Repository Status

```
Repository: volkanakbulut73/workigom
Branch: master
Latest Commit: ed54bb0
Remote URL: git@github.com:volkanakbulut73/workigom.git (SSH)
```

Files Added/Modified:

- ✓ railway.toml (project root)
- ✓ backend/Dockerfile (with marker comment)

Expected Outcome

After Railway rebuilds with these changes, you should see:

1. **In Build Logs:** The marker comment indicating correct Dockerfile usage
 2. **Build Process:** Backend-specific build steps (Node.js, Prisma, etc.)
 3. **No More Errors:** About nginx.conf or frontend-related files
 4. **Successful Deployment:** Backend API running and accessible
-

Need Help?

If you still encounter issues:

1. **Share the new build logs** from Railway
2. **Take screenshot** of Railway Settings showing:
 - Root Directory setting
 - Branch connected to production
 - Source Repo
3. **Check if** the marker comment appears in logs

The marker comment is your key indicator - if you see it, the Dockerfile configuration is working correctly.

Quick Verification Checklist

- [x] SSH connection to GitHub working
 - [x] railway.toml created and pushed
 - [x] Backend Dockerfile has marker comment
 - [x] Git remote using SSH
 - [x] Changes pushed to master branch
 - [] Railway rebuild triggered (next step for you)
 - [] Marker comment visible in Railway logs (verify after rebuild)
 - [] Build completing successfully (verify after rebuild)
 - [] Backend API accessible (verify after deployment)
-

Good luck with the deployment! The configuration is now correct on the GitHub side. The next steps are in Railway's dashboard. 🚀