

Backend/src/main.tsx Deletion Verification Summary

Date: October 24, 2025

Time: 21:15 UTC

Issue: Render.com still showing errors for deleted file `backend/src/main.tsx`

Verification Results

1. Local File System Check

```
ls -la backend/src/main.tsx
# Result: ls: cannot access 'backend/src/main.tsx': No such file or directory
```

Status:  File does NOT exist locally

2. Git Commit History

```
git log --oneline -5
```

Latest Commit: `c4139e1 fix: Remove frontend main.tsx from backend src directory`

- Committed: Fri Oct 24 21:10:24 2025 +0000

- Status:  File was properly deleted in commit `c4139e1`

3. Git Commit Details

```
commit c4139e13742815dce63b971e2e9592999b6ceee5
Date:   Fri Oct 24 21:10:24 2025 +0000
```

`fix: Remove frontend main.tsx from backend src directory`

```
backend/src/main.tsx | 7 -----
1 file changed, 7 deletions(-)
```

Status:  File deletion committed correctly

4. Remote Repository Check

```
git log origin/master -1
```

Result: Remote is at commit `c4139e1` (same as local)

Status:  Changes pushed to GitHub successfully

5. Git Tree Verification

```
git ls-tree -r HEAD --name-only | grep -i main.tsx
```

Result: Only `src-frontend/main.tsx` exists (correct location)

Status: No `main.tsx` in `backend/src` directory

6. Backend Directory Contents (Current HEAD)

```
git ls-tree -r HEAD --name-only backend/src/
```

Files in `backend/src`:

- `app.ts`
- `config/.ts`
- `controllers/.ts`
- `middleware/.ts`
- `routes/.ts`
- `server.ts`
- `types/.ts`
- `utils/.ts`
- **`main.tsx`** (NOT PRESENT - as expected)

7. File History

```
git log --all --full-history --oneline -- backend/src/main.tsx
```

Result:

- `f54b291` - File was added (moved to backend directory)
- `c4139e1` - File was deleted

Status: Clean deletion history

8. Local vs Remote Sync

```
git diff HEAD origin/master
```

Result: No differences

Status: Local and remote are in sync

🎯 Conclusion

ALL CHECKS PASSED

1. **File is deleted locally**
2. **Deletion was committed**
3. **Changes were pushed to GitHub**
4. **Remote repository is up to date**
5. **No `main.tsx` exists in `backend/src/`**



The Issue: Render.com Build Cache

Since the file is definitively deleted from GitHub and all checks pass, the error you're seeing is due to **Render.com's build cache**.

Why is Render still seeing the old file?

1. Render uses **Docker layer caching** to speed up builds
2. The old **COPY command cached** the file before it was deleted
3. Render needs to **rebuild from scratch** to see the changes



Solution: Force Render to Clear Cache

You have **THREE options** to fix this:

Option 1: Manual Redeploy (Recommended)

1. Go to your Render dashboard
2. Click on the `workigom` service
3. Click “**Manual Deploy**” → “**Clear build cache & deploy**”
4. This will force a fresh build from the latest commit

Option 2: Trigger Rebuild via Settings

1. In Render dashboard, go to service Settings
2. Scroll to “**Build & Deploy**”
3. Click “**Clear Build Cache**”
4. Then click “**Manual Deploy**”

Option 3: Add Cache Bust (Already Done)

Your Dockerfile already has:

```
ARG CACHE_BUST=202510232325
```

To force a new build, you can update this value:

```
# Update the cache bust value
sed -i 's/CACHE_BUST=.*/CACHE_BUST=$(date +%Y%m%d%H%M) /' backend/Dockerfile

# Commit and push
git add backend/Dockerfile
git commit -m "chore: Update cache bust to force rebuild"
git push origin master
```



Render Configuration Verification

Based on the screenshots provided:

Current Render Settings:

- **Repository:** volkanakbulut73/workigom
- **Branch:** master 
- **Root Directory:** backend/ 
- **Build Command:** Uses detected Dockerfile 

Recommendations:

1.  Settings are correct
 2.  Need to clear build cache
 3.  Wait for fresh deployment
-



Next Steps

1. **Clear Render's build cache** using Option 1 or 2 above
 2. **Monitor the new deployment** in Render dashboard
 3. **Check logs** to confirm the file is no longer being compiled
 4. **Verify the build succeeds** without TypeScript errors
-



Expected Outcome

After clearing the cache and redeploying:

-  No more errors about react-dom/client
 -  No more errors about ./App.tsx
 -  Backend TypeScript should compile successfully
 -  Deployment should complete successfully
-



Debug Info (If Issue Persists)

If you still see errors after clearing cache:

1. **Check the build logs** for the exact commit hash being used
 2. **Verify Render is using the correct branch** (master)
 3. **Check if Render is using a fork** or outdated webhook
 4. **Ensure no other services** are trying to build from old commits
-



Technical Summary

What Happened:

1. File backend/src/main.tsx was accidentally created in backend directory
2. This caused TypeScript compilation errors (React file in Node.js project)
3. File was properly deleted and pushed to GitHub

4. Render's build cache still contains the old file

What We Did:

1. Verified file deletion locally
2. Verified commit was created correctly
3. Verified push to GitHub succeeded
4. Verified remote repository is clean
5. Documented the cache issue

What You Need to Do:

1. **Clear Render's build cache**
 2. **Trigger a fresh deployment**
 3. **Monitor the build logs**
-

Generated: October 24, 2025 21:15 UTC

Status:  All verification checks passed - Issue is Render-side caching